Making Middleboxes Someone Else's Problem: Network Processing as a Cloud Service

Justine Sherry
UC Berkeley

Shaddi Hasan
UC Berkeley

Colin Scott
UC Berkeley

Arvind Krishnamurthy
University of Washington

Sylvia Ratnasamy
UC Berkeley

Vyas Sekar
Intel Labs

Austin Wang, Mike He, Kaiqu Liang, Haichen Dong

COS 561

Feb 24 2023

Topics

Intro and Background: What are middleboxes Sec 1&2 - Austin
Sec. 3 and/or 4: Design and Implementation - Mike
Sec. 5: Evaluation - Haichen
Sec. 6 and 7: Discussion and related work - Kaiqu

What are Middleboxes

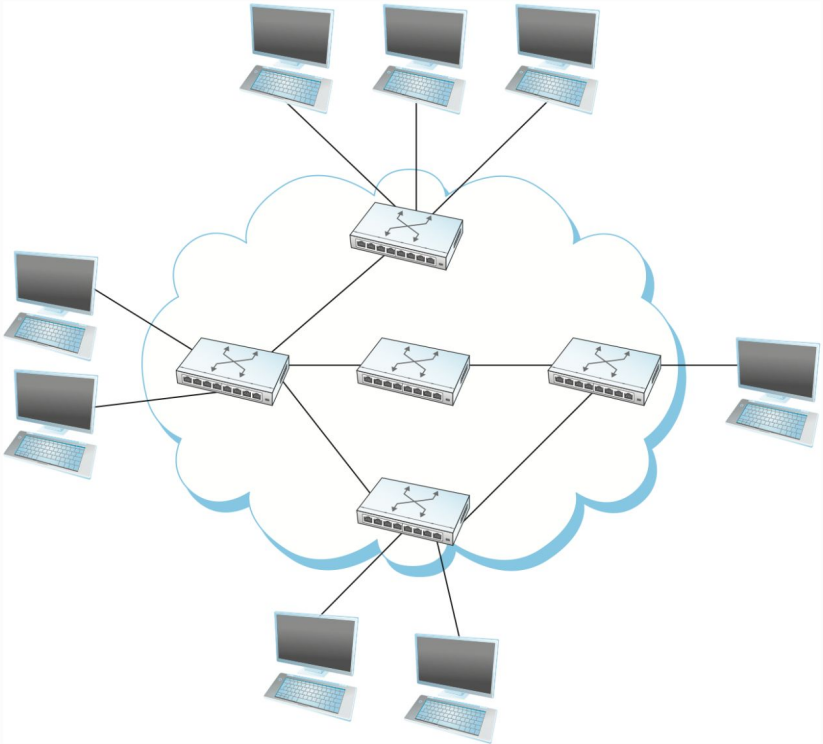


Figure 3. *Switched network.*

From *Computer Networks: A Systems Approach*

What are Middleboxes

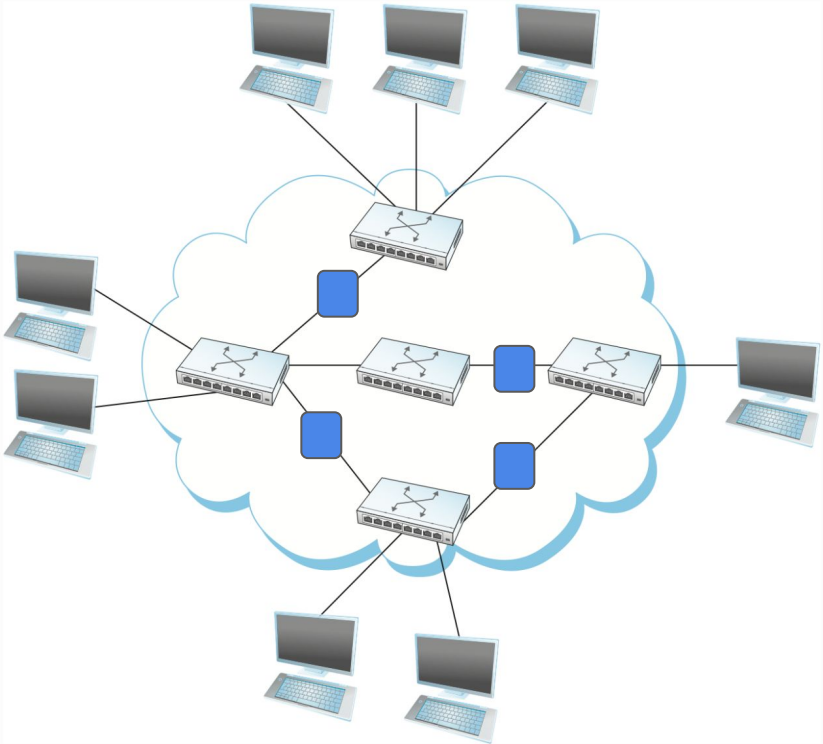


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

From *Computer Networks: A Systems Approach*

What are Middleboxes

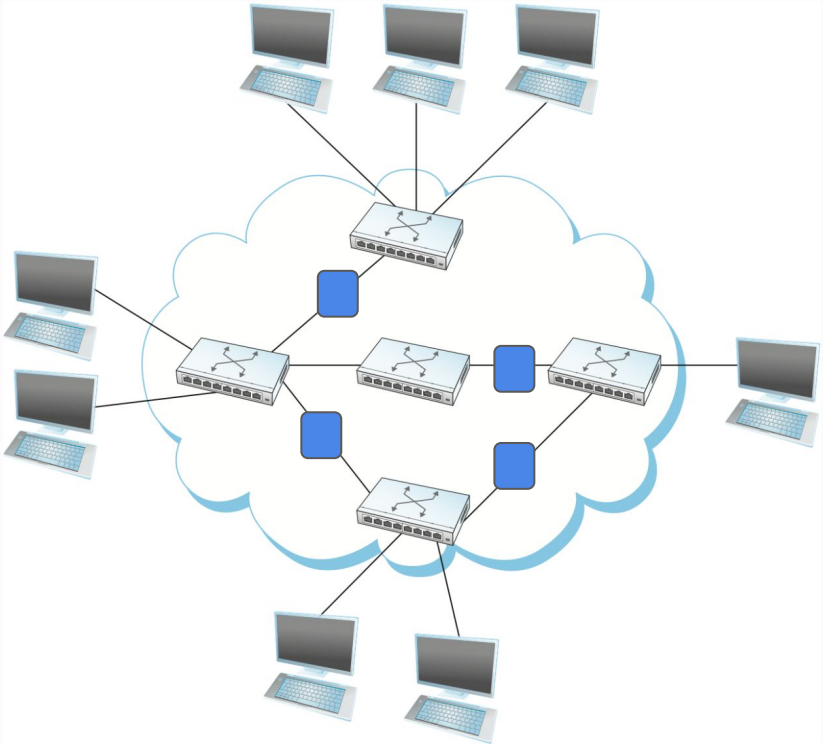


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- Security
- Performance

From *Computer Networks: A Systems Approach*

Middlebox Examples: Security

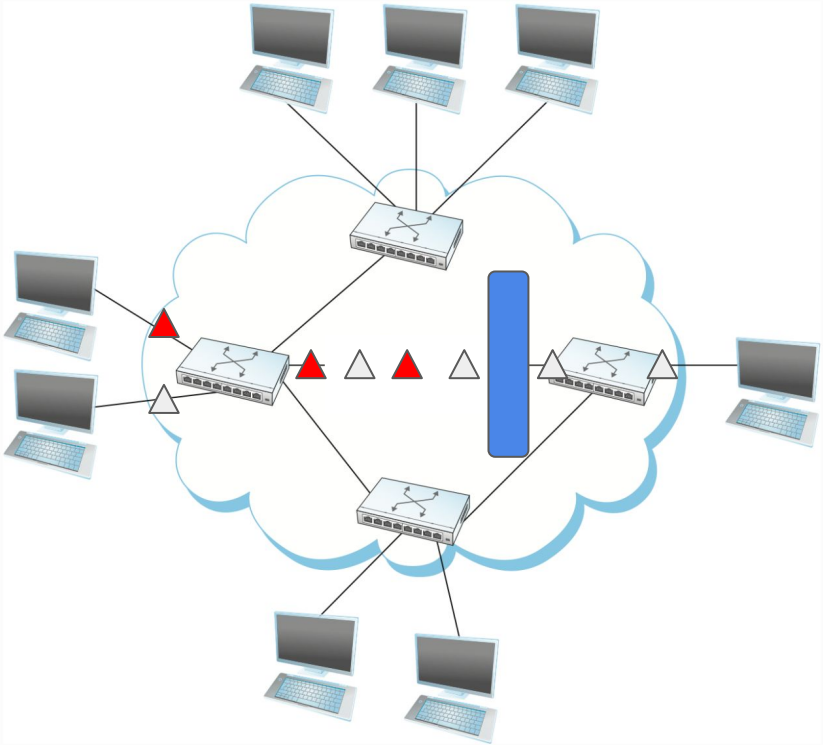


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

Security

- **Firewall:** Filter traffic based on security rules (e.g. disallow traffic to certain ports)

From *Computer Networks: A Systems Approach*

Middlebox Examples: Security

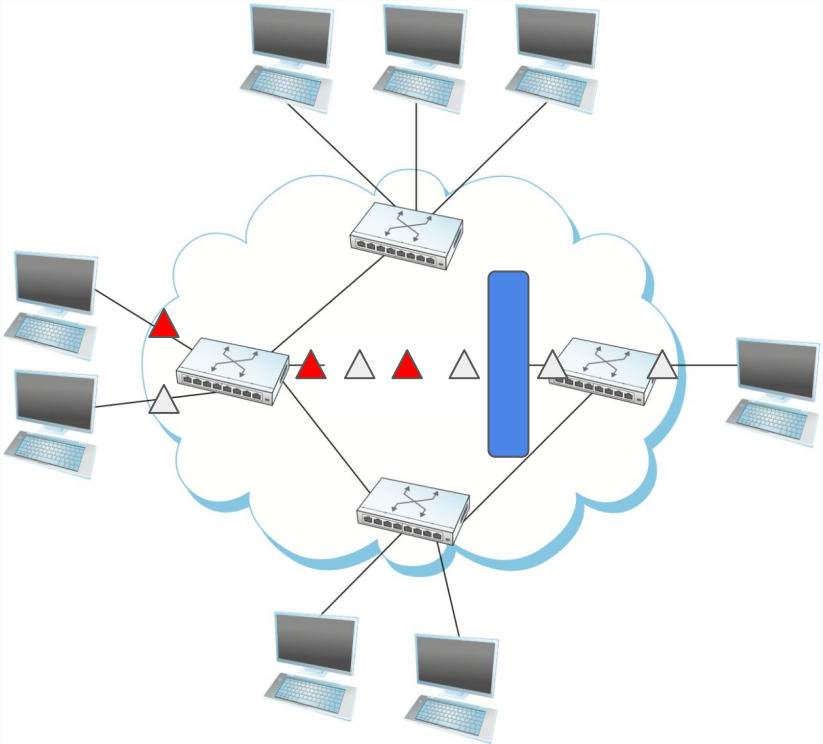


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

Security

- **Firewall:** Filter traffic based on security rules (e.g. disallow traffic to certain ports)

- **Intrusion Detection Systems:** Monitor traffic and collect data for offline analysis of security anomalies.

From *Computer Networks: A Systems Approach*

Middlebox Examples: Performance

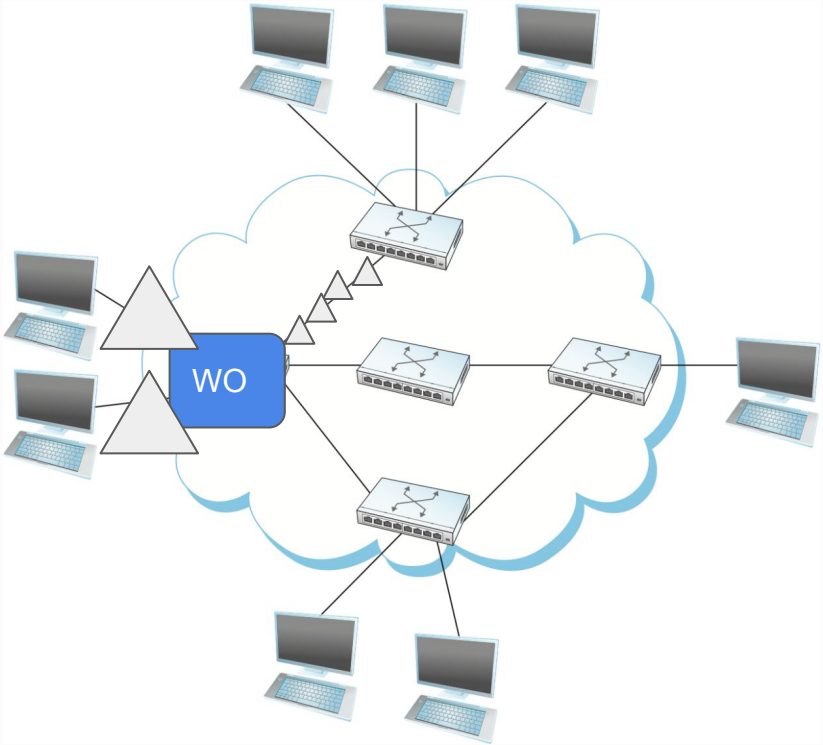


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

Performance

- **WAN Optimizers:** Optimizer network traffic by compression / deduplication / caching etc.

From *Computer Networks: A Systems Approach*

Middlebox Examples: Performance

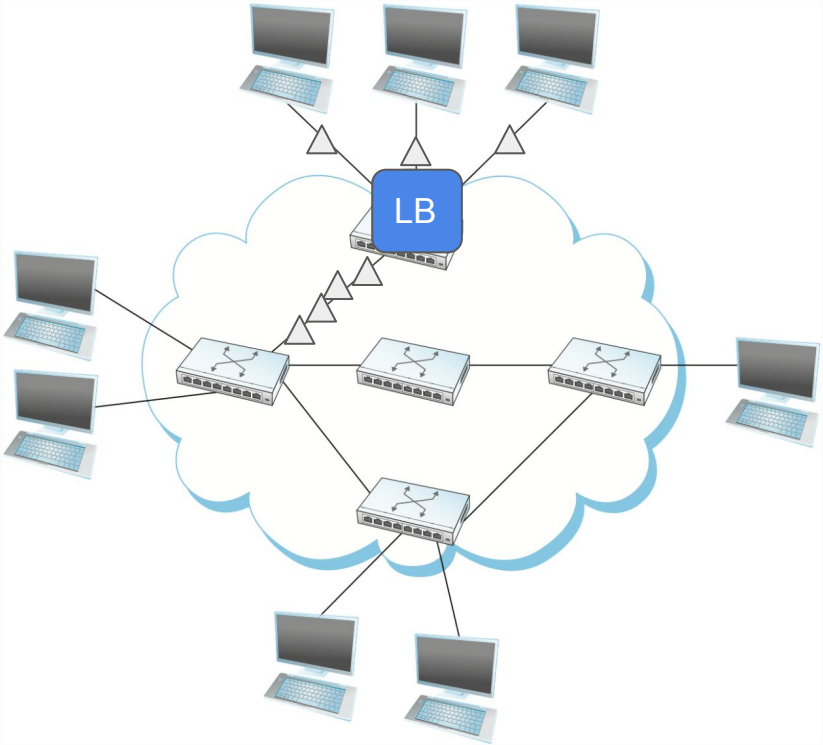


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

Performance

- **WAN Optimizers:** Optimizer network traffic by compression / deduplication / caching etc.
- **Load Balancer:** Forward traffic flow to one or more hosts

From *Computer Networks: A Systems Approach*

What are Middleboxes

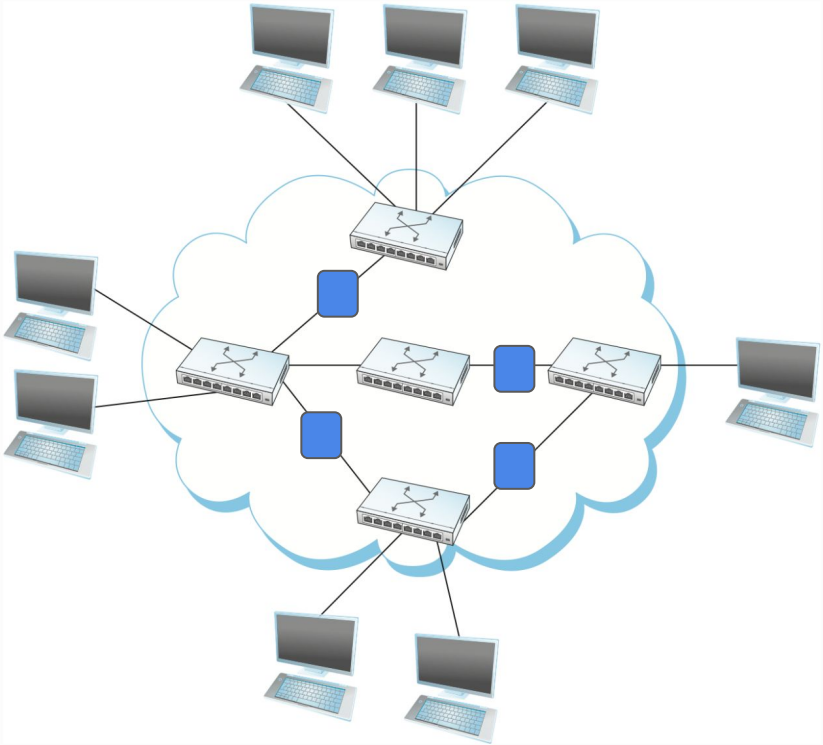


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- Security
- Performance

Can be implemented with dedicated hardware:

From *Computer Networks: A Systems Approach*

What are Middleboxes

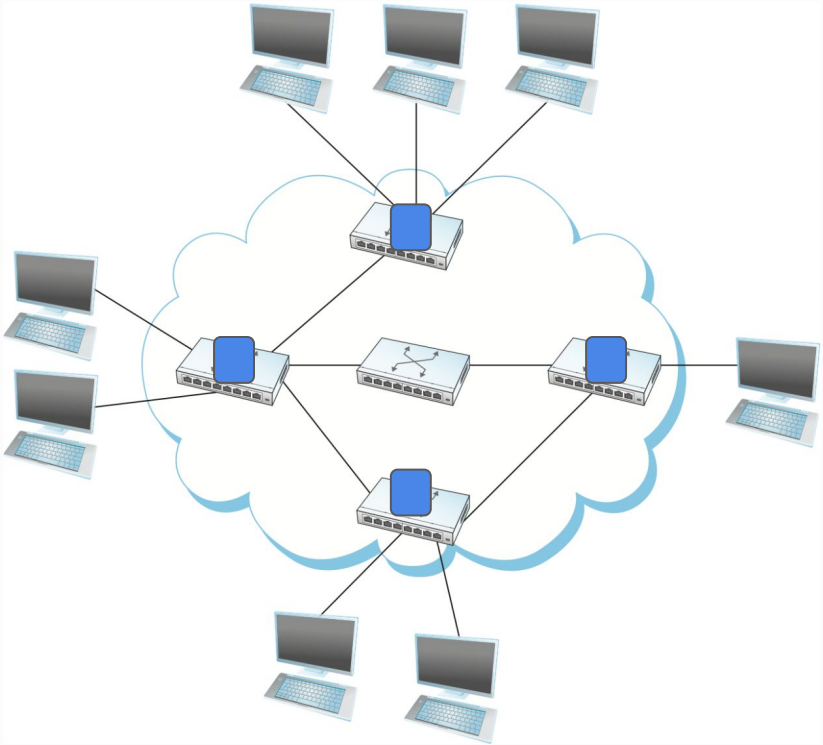
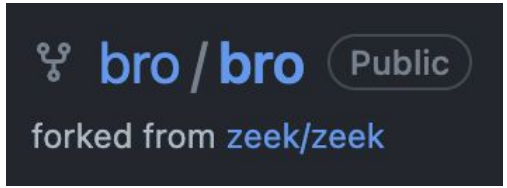


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- Security
- Performance

Can be implemented in software and run on commodity hardware:



From *Computer Networks: A Systems Approach*

Middleboxes: Challenges

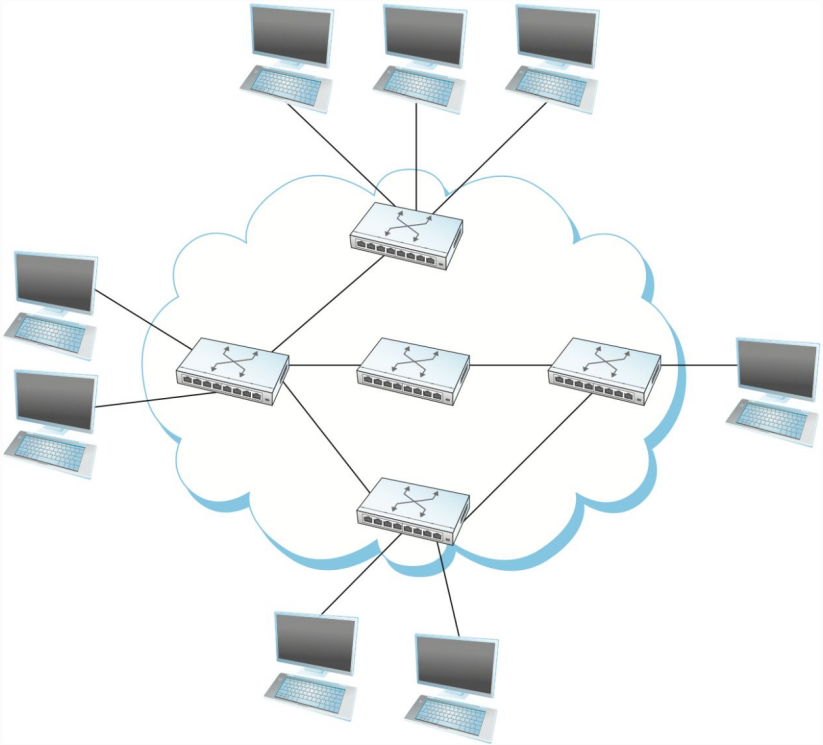


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

From *Computer Networks: A Systems Approach*

Middleboxes: Challenges

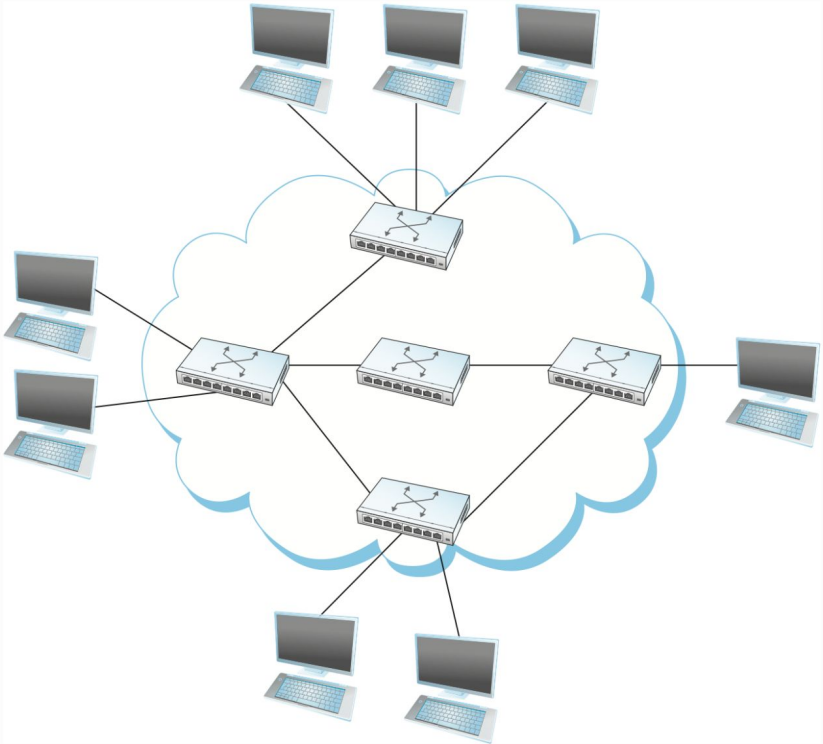


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- **E2E:** Middleboxes violate the E2E Principle: Features such as reliability and security should be implemented at the endpoints.

From *Computer Networks: A Systems Approach*

Middleboxes: Challenges

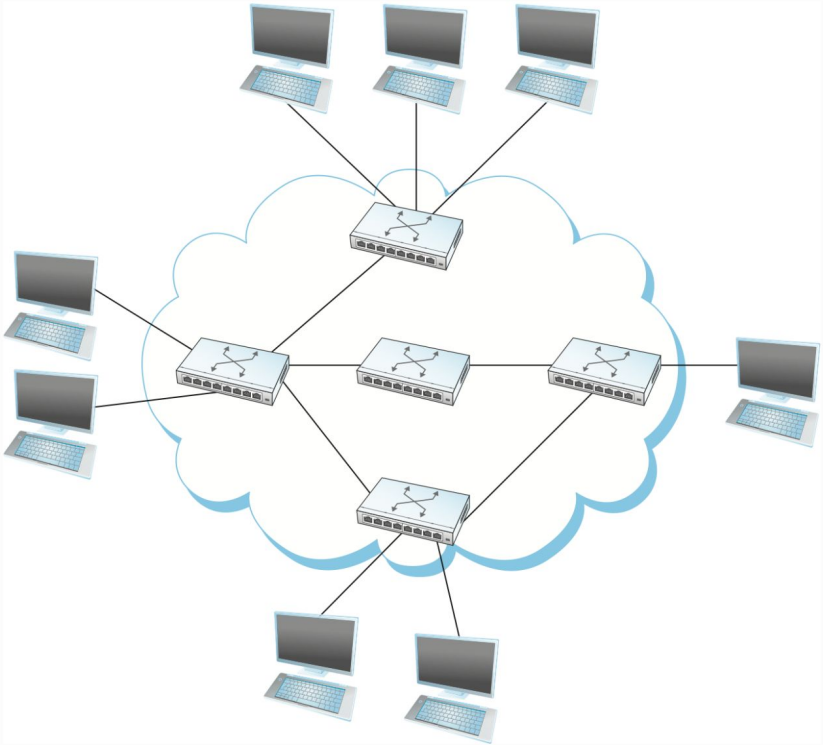


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- **E2E:** Middleboxes violate the E2E Principle: Features such as reliability and security should be implemented at the endpoints.
- **Compatibility:** If all my middleboxes assume HTTP1.5 hard to upgrade to HTTP2

From *Computer Networks: A Systems Approach*

Middleboxes: Challenges

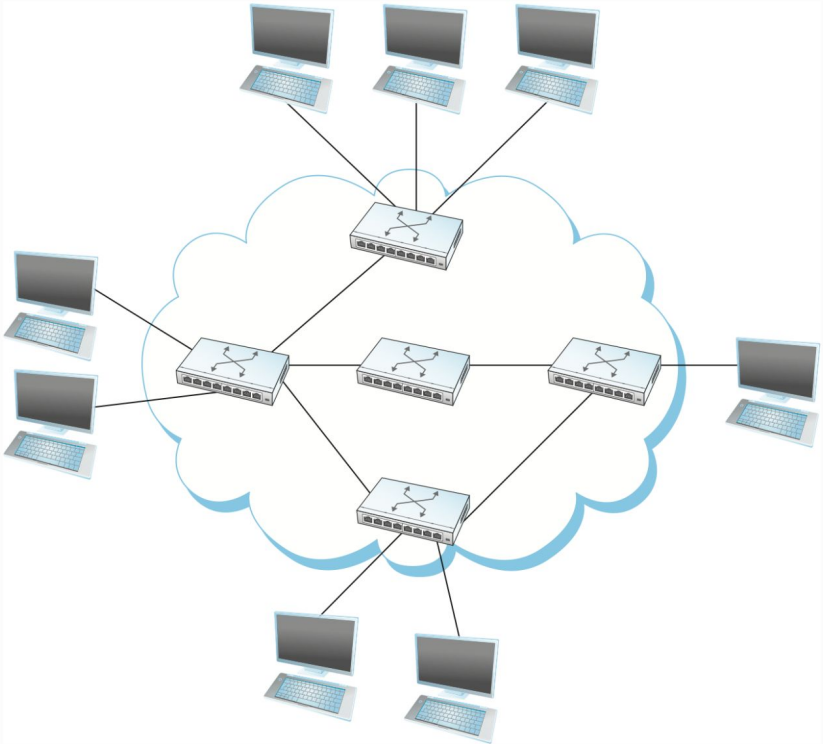


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- **E2E:** Middleboxes violate the E2E Principle: Features such as reliability and security should be implemented at the endpoints.
- **Compatibility:** If all my middleboxes assume HTTP1.5 hard to upgrade to HTTP2
- **Statefulness:** Middleboxes may have to be stateful (e.g. IDS must store data offline)

From *Computer Networks: A Systems Approach*

Middleboxes: Challenges

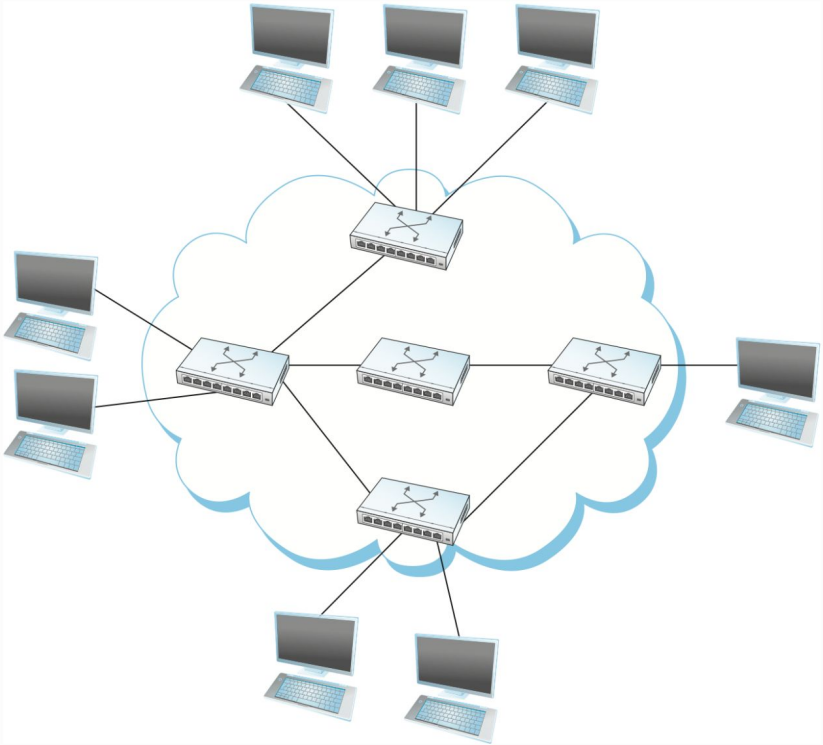


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- **E2E:** Middleboxes violate the E2E Principle: Features such as reliability and security should be implemented at the endpoints.
- **Compatibility:** If all my middleboxes assume HTTP1.5 hard to upgrade to HTTP2
- **Statefulness:** Middleboxes may have to be stateful (e.g. IDS must store data offline)
- **Privacy:** More points of attack?

From *Computer Networks: A Systems Approach*

Middleboxes: Challenges

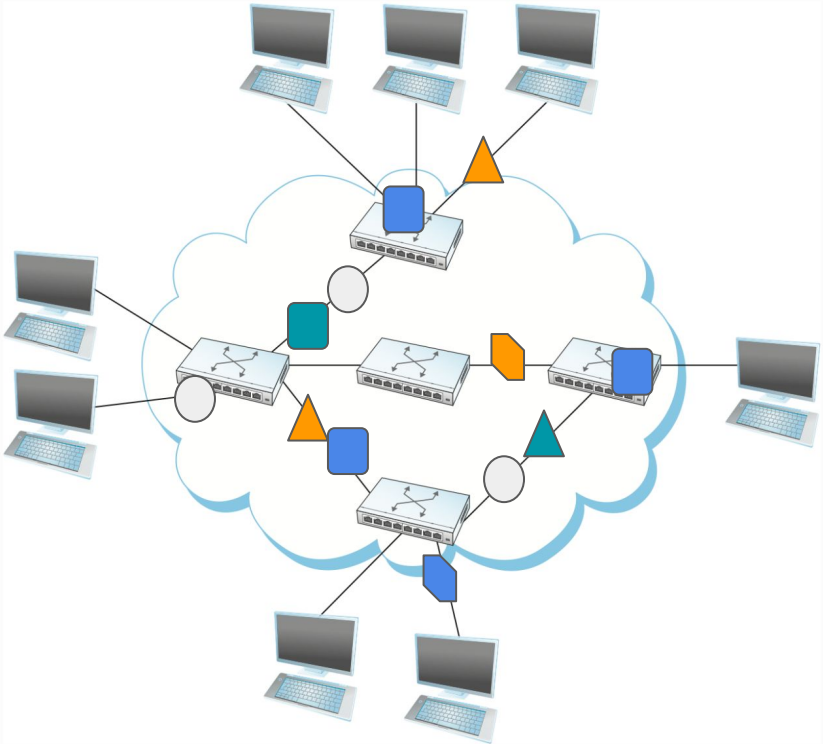


Figure 3. *Switched network.*

Middleboxes are intermediary devices between source and destination host that performs functions **other than packet forwarding.**

- Despite the challenges, increasing use of middleboxes in networks
- Increasingly difficult to manage!

From *Computer Networks: A Systems Approach*

APLOMB: Core Idea

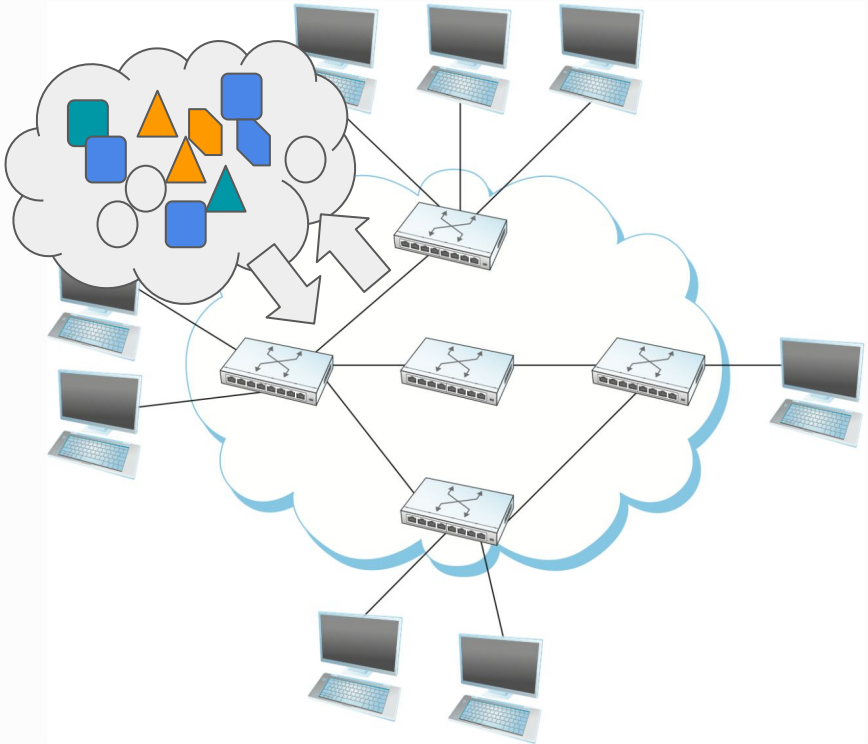


Figure 3. *Switched network.*

Can we move middlebox services to the cloud?

From *Computer Networks: A Systems Approach*

APLOMB: Core Idea

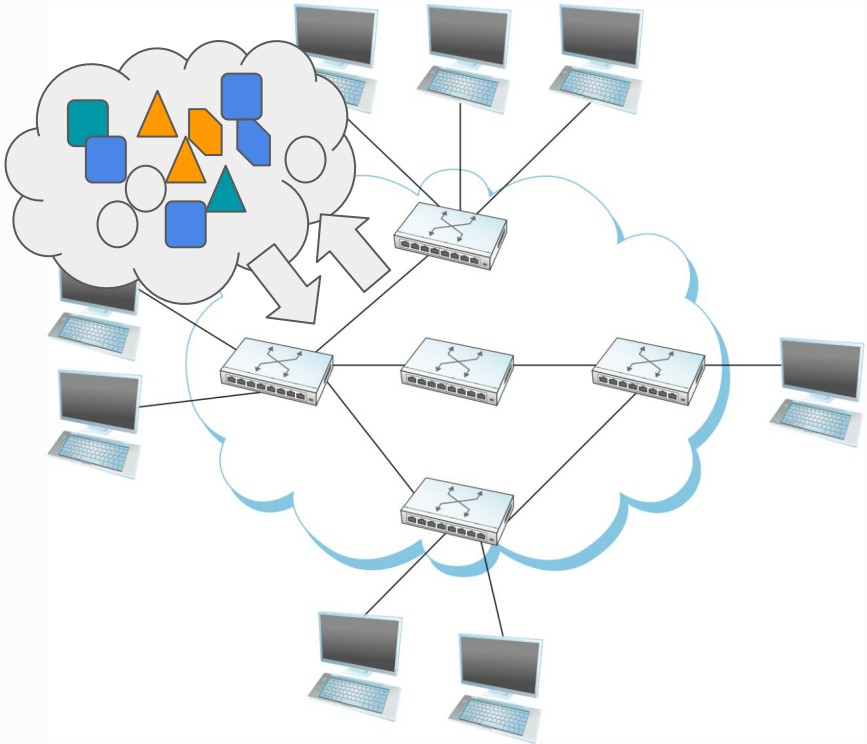


Figure 3. *Switched network.*

From *Computer Networks: A Systems Approach*

Can we move middlebox services to the cloud?

1. Is there a need? What does enterprise middlebox infrastructure look like today?
    a. Deployment
    b. Management
    c. Failure cases

APLOMB: Core Idea

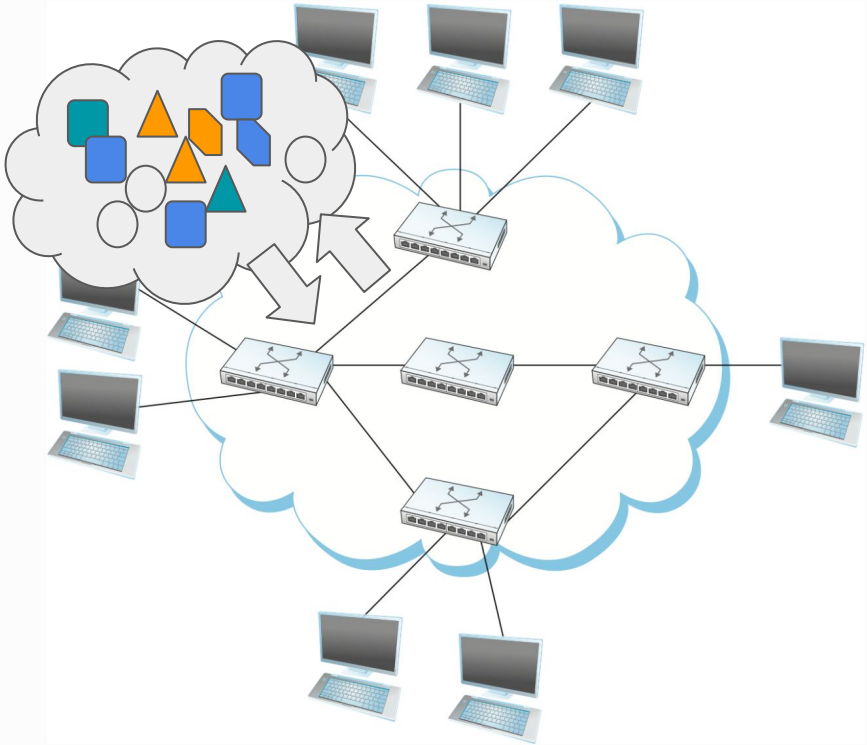


Figure 3. *Switched network.*

From *Computer Networks: A Systems Approach*

Can we move middlebox services to the cloud?

1. Is there a need? What does enterprise middlebox infrastructure look like today?
    a. Deployment
    b. Management
    c. Failure cases
2. What requirements would a cloud middlebox service need to satisfy?

Middleboxes Today

Survey of 57 enterprise network admins:

- 19 small (< 1k hosts) networks
- 18 medium (1k - 10k hosts) networks
- 11 large (10k - 100k hosts) networks
- 7 very large (> 100k hosts) networks

Middleboxes Today: Deployment

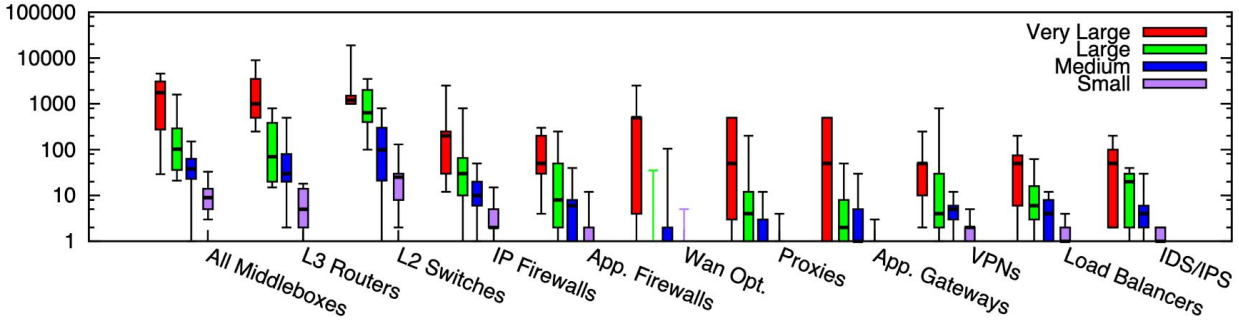


Figure 1: Box plot of middlebox deployments for small (fewer than 1k hosts), medium (1k-10k hosts), large (10k-100k hosts), and very large (more than 100k hosts) enterprise networks. Y-axis is in log scale.

Average very large network:

- ~2800 L3 routers
- ~1900 middleboxes

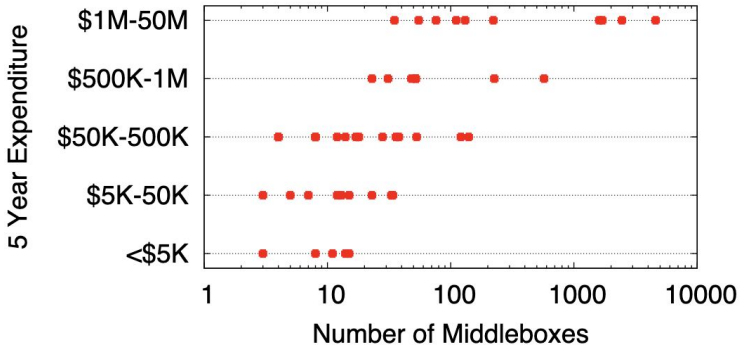


Figure 2: Administrator-estimated spending on middlebox hardware per network.

Middleboxes Today: Management

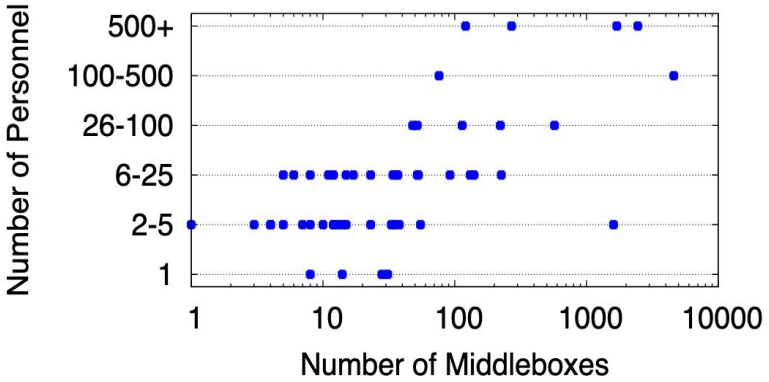


Figure 3: Administrator-estimated number of personnel per network.

Middleboxes Today: Management

Complexities in management + how cloud solution can help

1. **Upgrades:** Purchasing and upgrading hardware is time-consuming and locks admins in to hardware.
    a. In cloud solution, hardware upgrades are abstracted away, enterprises sign up for *service*

Middleboxes Today: Management

Complexities in management + how cloud solution can help

1. **Upgrades:** Purchasing and upgrading hardware is time-consuming and locks admins in to hardware.
    a. In cloud solution, hardware upgrades are abstracted away, enterprises sign up for *service*

2. **Monitoring:** Middleboxes need to be monitored for failures
    a. Cloud providers monitors utilization and failures

Middleboxes Today: Management

Complexities in management + how cloud solution can help

1. **Upgrades:** Purchasing and upgrading hardware is time-consuming and locks admins in to hardware.
   a. In cloud solution, hardware upgrades are abstracted away, enterprises sign up for *service*
2. **Monitoring:** Middleboxes need to be monitored for failures
   a. Cloud providers monitors utilization and failures
3. **Configuration:** Need to configure appliance to ensure proper operation
   a. Cloud providers responsible for low-level appliance configuration

Middleboxes Today: Management

Complexities in management + how cloud solution can help

1. **Upgrades:** Purchasing and upgrading hardware is time-consuming and locks admins in to hardware.
   a. In cloud solution, hardware upgrades are abstracted away, enterprises sign up for *service*
2. **Monitoring:** Middleboxes need to be monitored for failures
   a. Cloud providers monitors utilization and failures
3. **Configuration:** Need to configure appliance to ensure proper operation
   a. Cloud providers responsible for low-level appliance configuration
4. **Training:** New appliances (1) require training admins to manage them.
   a. Many admin tasks are handled by the cloud provider

Middleboxes Today: Failures

	Misconfig.	Overload	Physical/Electric
Firewalls	67.3%	16.3%	16.3%
Proxies	63.2%	15.7%	21.1%
IDS	54.5%	11.4%	34%

Table 1: Fraction of network administrators who estimated misconfiguration, overload, or physical/electrical failure as the most common cause of middlebox failure.

Requirements of a Cloud Middlebox

1. **Functional Equivalence:** Cloud-based middlebox must offer functionality and semantics equivalent to on-site middlebox.
    a. Certain solutions that require the middlebox to be “on path” may be hard to implement

Requirements of a Cloud Middlebox

1. **Functional Equivalence:** Cloud-based middlebox must offer functionality and semantics equivalent to on-site middlebox.
    a. Certain solutions that require the middlebox to be “on path” may be hard to implement

2. **Low complexity at the enterprise:** Cloud-based middlebox requires some way for the enterprise to connect to the service. The cost and complexity of such connection schemes should be as low as possible.

Requirements of a Cloud Middlebox

1. **Functional Equivalence:** Cloud-based middlebox must offer functionality and semantics equivalent to on-site middlebox.
   a. Certain solutions that require the middlebox to be “on path” may be hard to implement

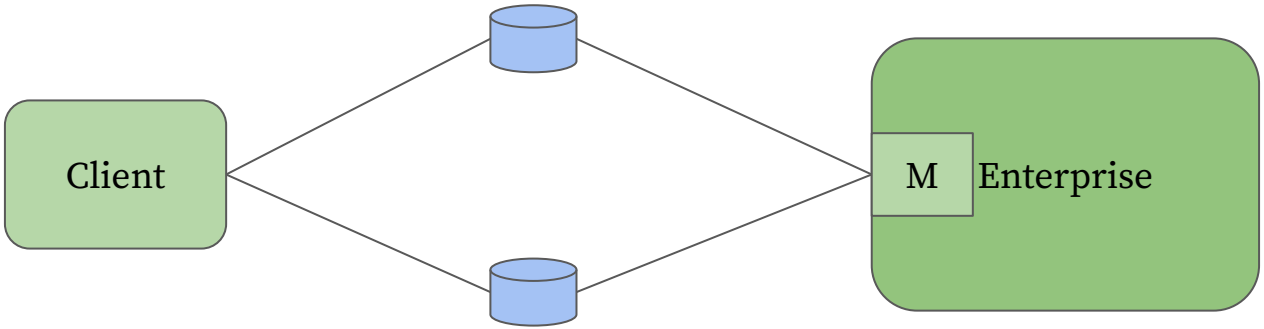
2. **Low complexity at the enterprise:** Cloud-based middlebox requires some way for the enterprise to connect to the service. The cost and complexity of such connection schemes should be as low as possible.

3. **Low performance overhead:** Cloud-based middleboxes requires detours, the latency and bandwidth consumption of the detour should be minimized.

Design Space

Cause of APLOMB: eliminating 3 key properties of "today's" ~~(decades ago)~~: middleboxes are

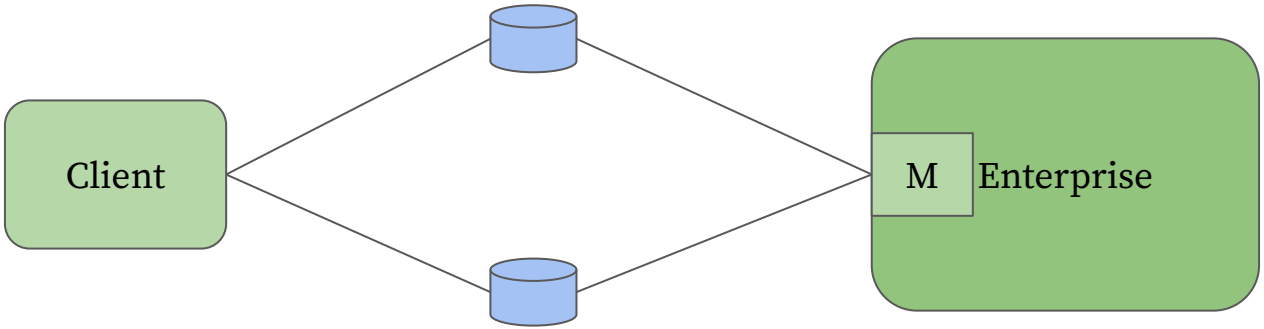
1. *On-path:* lie on the direct IP path of the endpoints
2. *Choke-points:* all paths between a pair of endpoints
3. *Local:* present in enterprises



Design Space

Cause of APLOMB: eliminating 3 key properties of "today's" ~~(decades ago)~~: middleboxes are

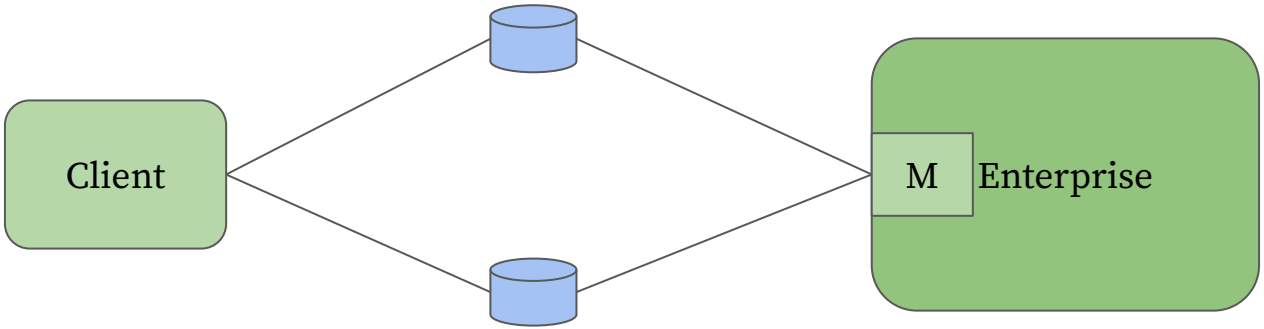
1. *On-path:* lie on the direct IP path of the endpoints
   a. Easy to obtain traffic they need to handle
   b. Traffic from each side is visible
2. *Choke-points:* all paths between a pair of endpoints
3. *Local:* present in enterprises



Design Space

Cause of APLOMB: eliminating 3 key properties of "today's" ~~(decades ago)~~: middleboxes are

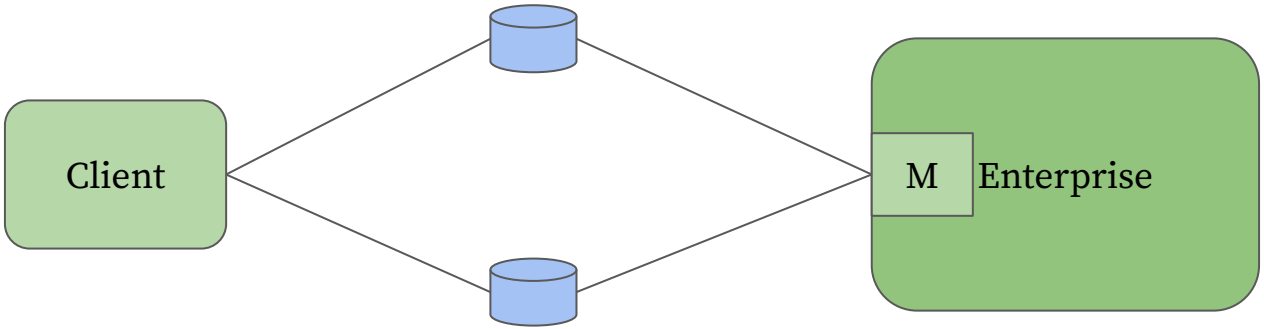
1. *On-path:* lie on the direct IP path of the endpoints
2. *Choke-points:* all paths between a pair of endpoints
   a. Traffic drive thru; no additional latency
3. *Local:* present in enterprises



Design Space

Cause of APLOMB: eliminating 3 key properties of "today's" ~~(decades ago)~~: middleboxes are

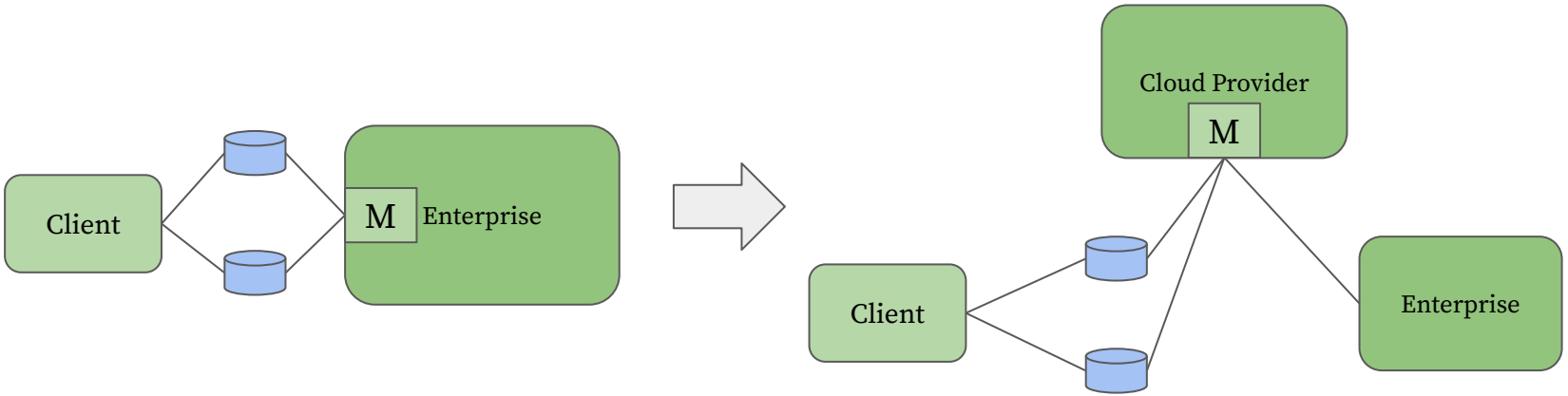
1. *On-path:* lie on the direct IP path of the endpoints
2. *Choke-points:* all paths between a pair of endpoints
3. *Local:* present in enterprises
    a. Necessary for location-dependent middleboxes, e.g. traffic compression



Design Space

Cause of APLOMB: eliminating 3 key properties of "today's" ~~(decades ago)~~: middleboxes are

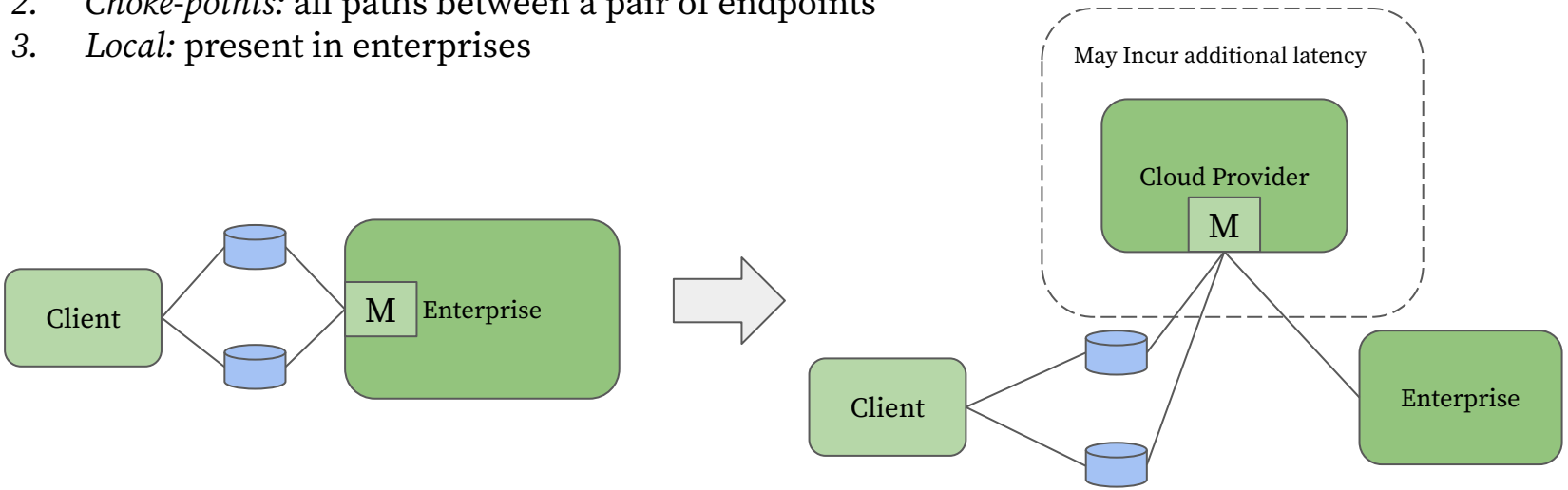
1. *On-path:* lie the direct IP path of the endpoints
2. *Choke-points:* all paths between a pair of endpoints
3. *Local:* present in enterprises



Design Space

Cause of APLOMB: eliminating 3 key properties of "today's" ~~(decades ago)~~: middleboxes are

1. *On-path:* lie the direct IP path of the endpoints
2. *Choke-points:* all paths between a pair of endpoints
3. *Local:* present in enterprises



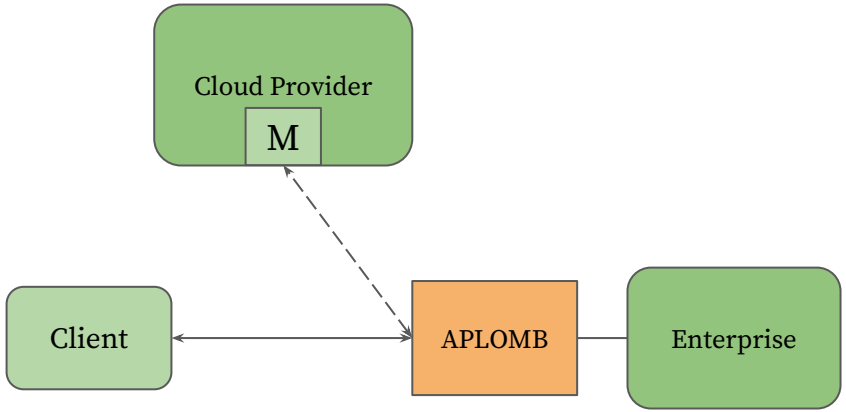
Design Space

Raises 3 questions about the (possible) designs

1. How to redirect the traffic s.t. increase in latency is minimized?
2. What type(s) of the provider's footprint is necessary for reducing the latency?
3. What type(s) of the middleboxes can be outsourced? What kind of functionality need to remain?

Design Space - Redirection Scheme

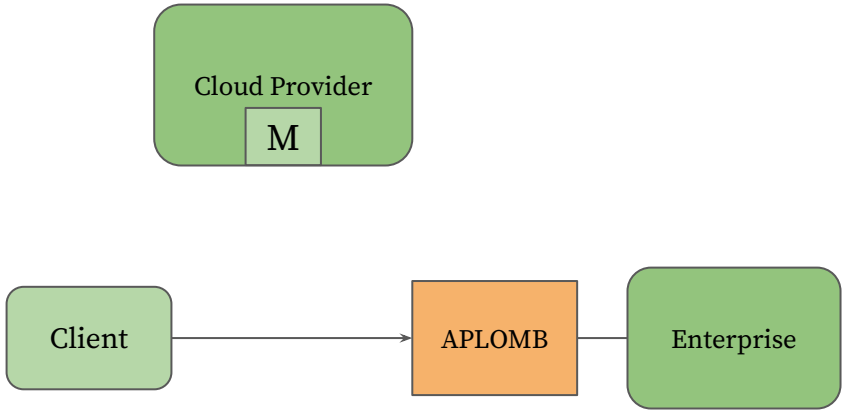
Bounce Redirection



Design Space - Redirection Scheme

Bounce Redirection

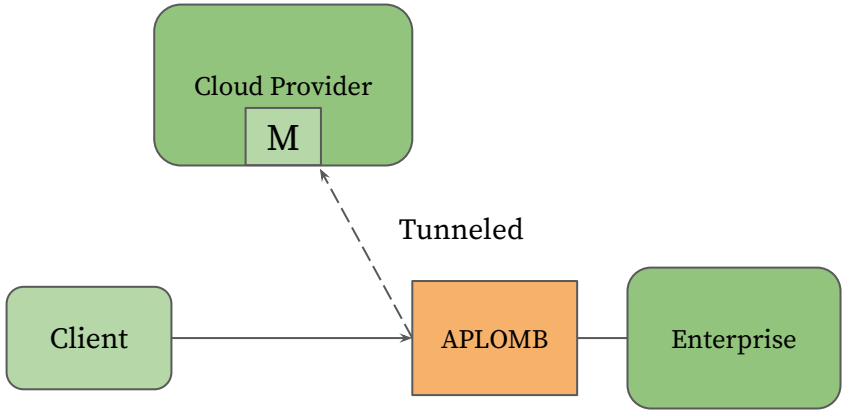
In-bound traffic



Design Space - Redirection Scheme

Bounce Redirection

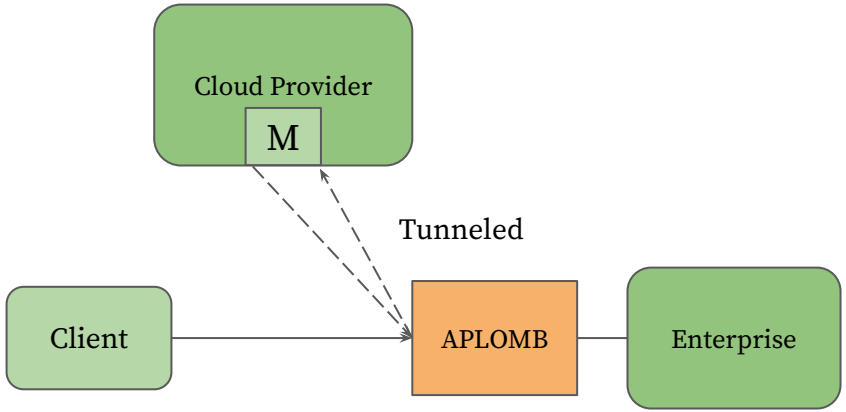
In-bound traffic



Design Space - Redirection Scheme

Bounce Redirection

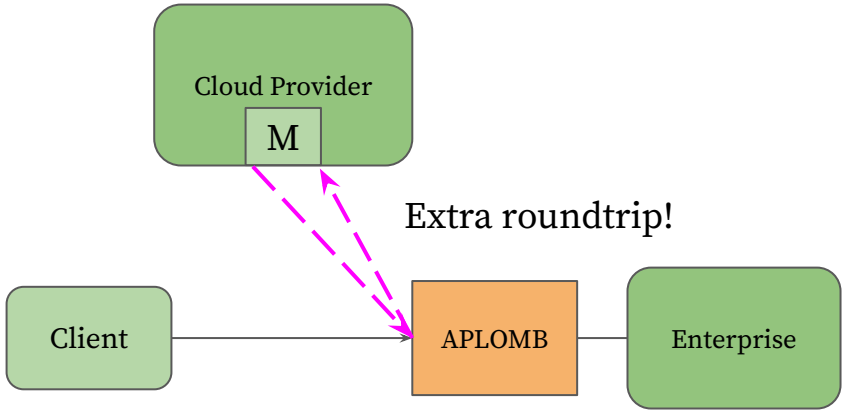
In-bound traffic



Design Space - Redirection Scheme

Bounce Redirection

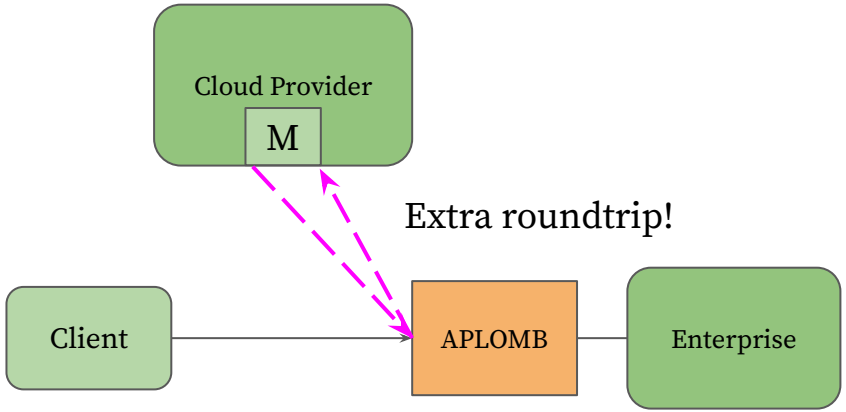
In-bound traffic



Design Space - Redirection Scheme

Bounce Redirection

In-bound traffic



Works under certain circumstances:
The cloud provider has a large footprint (i.e. the service is largely available across the Internet in different regions)

Design Space - Redirection Scheme

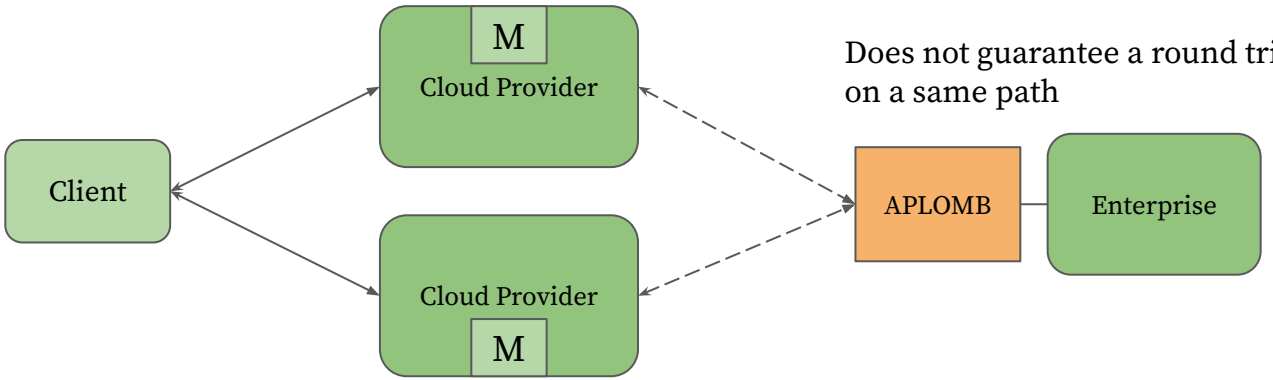
IP Redirection

Cloud Provider release IP prefix on the Enterprise's behalf

Client directs their traffic to the Cloud Provider

Multi-PoP (point of presence)

Does not guarantee a round trip on a same path



Design Space - Redirection Scheme

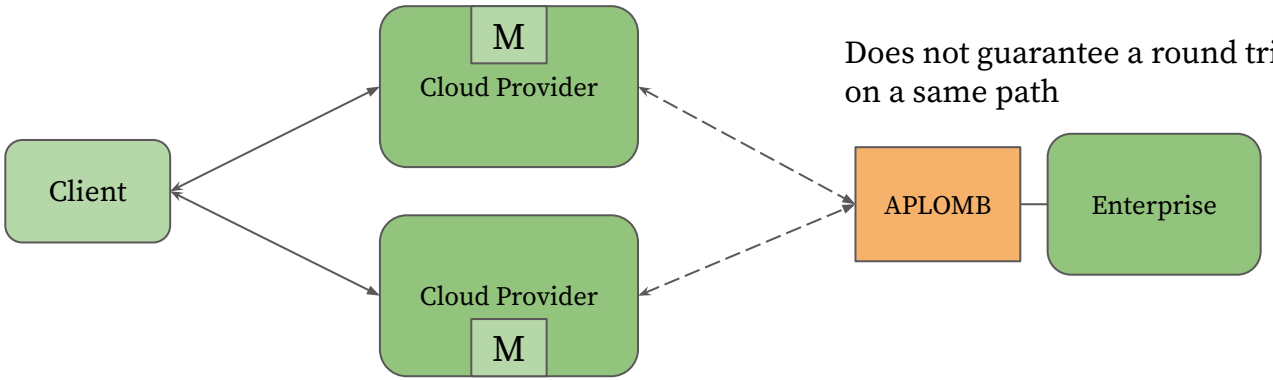
IP Redirection

Cloud Provider release IP prefix on the Enterprise's behalf

Client directs their traffic to the Cloud Provider

Multi-PoP (point of presence)

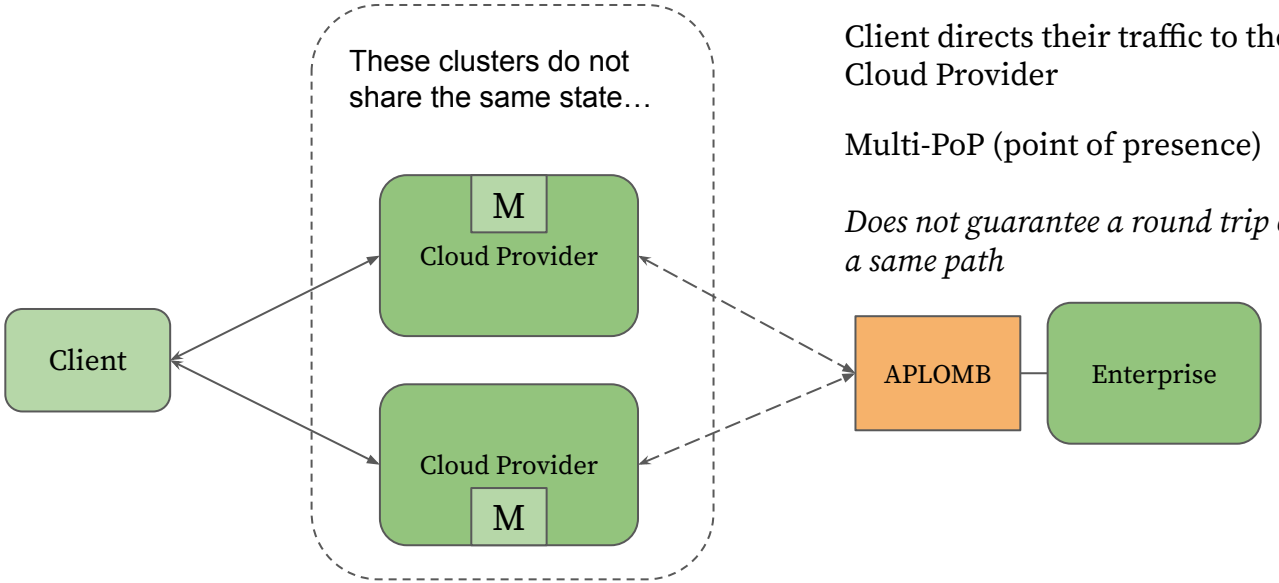
Does not guarantee a round trip on a same path



What about *stateful* middleboxes?

Design Space - Redirection Scheme

IP Redirection



Cloud Provider release IP prefix on the Enterprise's behalf

Client directs their traffic to the Cloud Provider

Multi-PoP (point of presence)

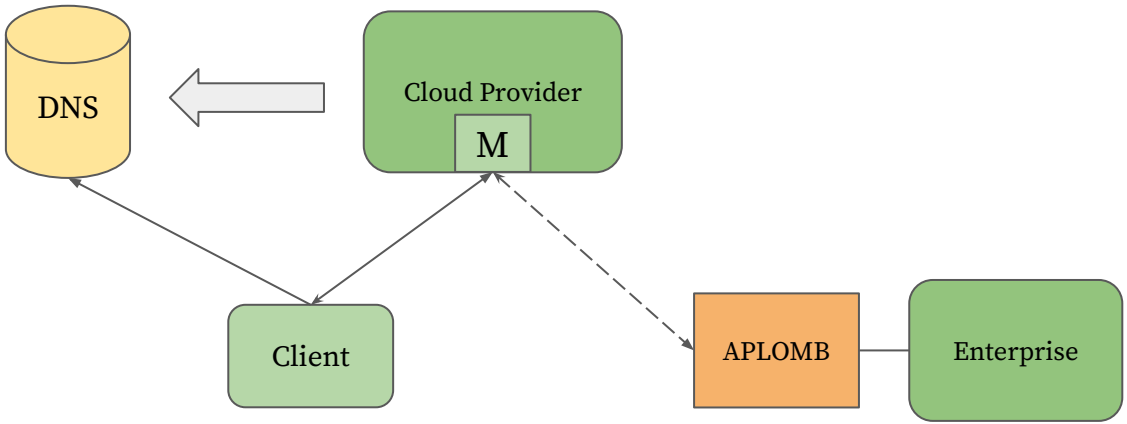
Does not guarantee a round trip on a same path

What about *stateful* middleboxes?

Design Space - Redirection Scheme

DNS Redirection (Adopted by APLOMB)

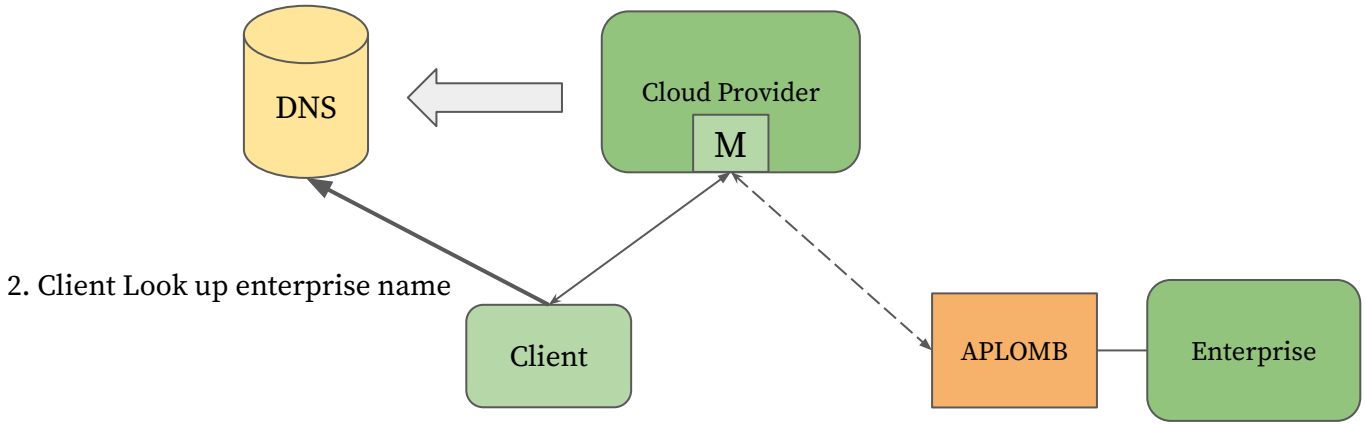
1. Register name look-up (Enterprise name → Cloud Provider address)



Design Space - Redirection Scheme

DNS Redirection (Adopted by APLOMB)

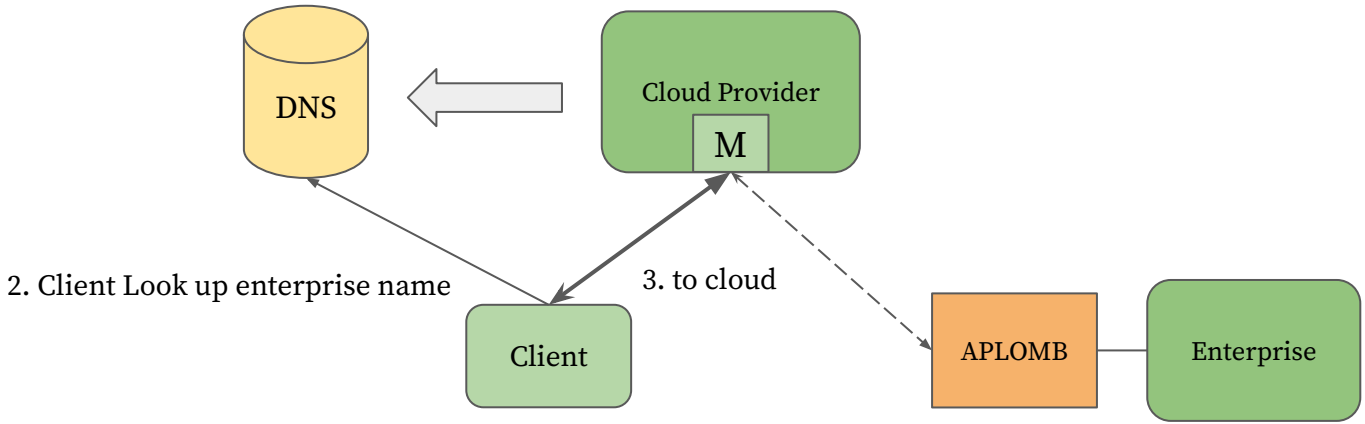
1. Register name look-up (Enterprise name → Cloud Provider address)



Design Space - Redirection Scheme

DNS Redirection (Adopted by APLOMB)

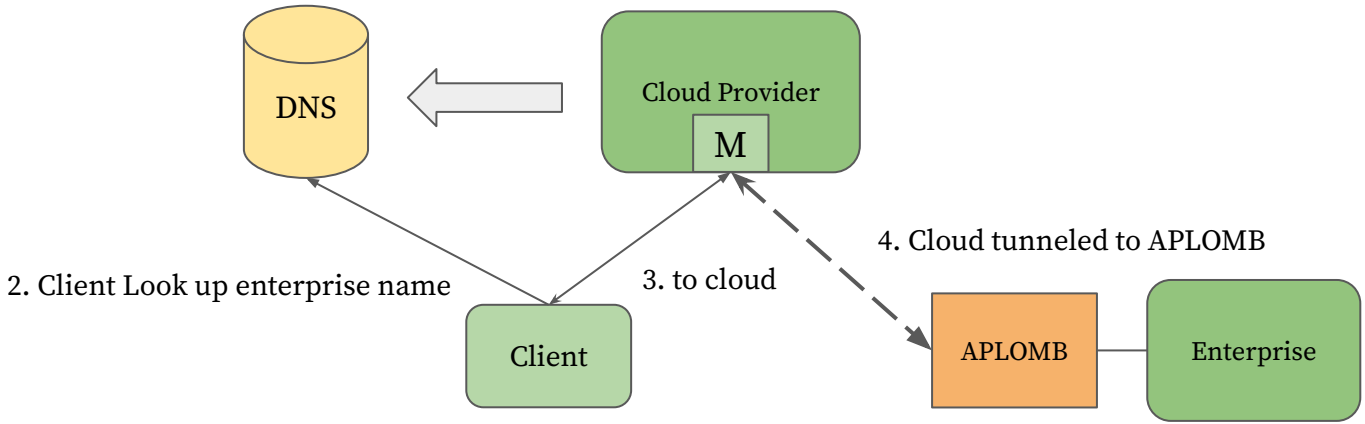
1. Register name look-up (Enterprise name → Cloud Provider address)



Design Space - Redirection Scheme

DNS Redirection (Adopted by APLOMB)

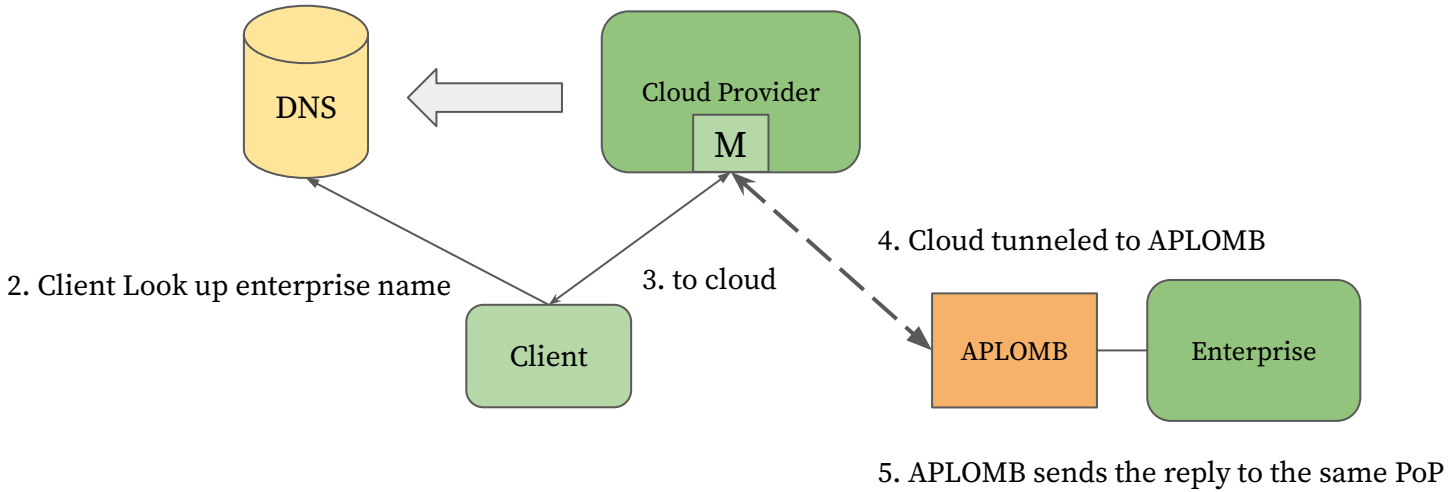
1. Register name look-up (Enterprise name → Cloud Provider address)



Design Space - Redirection Scheme

DNS Redirection (Adopted by APLOMB)

1. Register name look-up (Enterprise name → Cloud Provider address)



Design Space - Redirection Scheme

Discussion:

- Why we could not let the APLOMB in *IP Redirection* do the same thing as it in *DNS Redirection* to make the round trip on the same path?
- What additional failure could happen if we outsource middleboxes (using DNS Redirection) from the enterprise to the cloud? (i.e. parts that won't fail but now could possibly fail and cause availability issue)

Design Space - Redirection Scheme

Minimizing Latency

Latency may change depending on the choice of cloud providers. Naturally, we want to minimize the latency between the cloud provider and the enterprise, hence:

$$P^* = \arg\min_P(\text{Latency}(P, e))$$

Design Space - Redirection Scheme

Minimizing Latency

Latency may change depending on the choice of cloud providers. Naturally, we want to minimize the latency between the cloud provider and the enterprise, hence:

$$P^* = \arg\min_P(\text{Latency}(P, e))$$

Further, due to the triangle inequality violation, a better estimation could be

$$P^* = \arg\min_P(\text{Latency}(P, e) + \text{Latency}(P, c))$$

i.e. minimizing the sum of latencies from the cloud to client and to the enterprise

Design Space - Redirection Scheme

Minimizing Latency

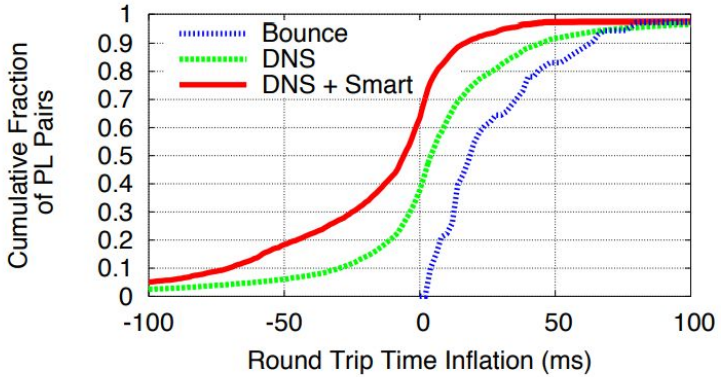


Figure 6: Round Trip Time (RTT) inflation when redirecting traffic between US PlanetLab nodes through Amazon PoPs.

Design Space - Provider Footprint

Footprint: the span of the provider network
Amazon-like: smaller footprint; well-connected (Multi-PoP)
Akamai-like: large footprint; better coverage (CDN)

Design Space - Provider Footprint

Footprint: the span of the provider network
Amazon-like: smaller number of the cloud centers; well-connected (Multi-PoP)
Akamai-like: large number of cloud centers; better coverage (CDN)

E.g. when sending a request to an Amazon server, we are directing our request to a specific server in a data center, maybe on the other half of the globe.

Design Space - Provider Footprint

Footprint: the span of the provider network
Amazon-like: smaller number of the cloud centers; well-connected (Multi-PoP)
Akamai-like: large number of cloud centers; better coverage (CDN)

E.g. when sending a request to an Amazon server, we are directing our request to a specific server in a data center, maybe on the other half of the globe.

On the other hand, when requesting some resource from an Akamai server, our request may be redirected to the one geographically proximate to us.

Design Space - Provider Footprint

Footprint: the span of the provider network
Amazon-like: smaller number of the cloud centers; well-connected (Multi-PoP)
Akamai-like: large number of cloud centers; better coverage (CDN)

E.g. when sending a request to an Amazon server, we are directing our request to a specific server in a data center, maybe on the other half of the globe.

On the other hand, when requesting some resource from an Akamai server, our request may be redirected to the one geographically proximate to us.

This might not be available for Amazon service due to their smaller *footprint* (i.e. there might not be an available server near us geographically)

Design Space - Provider Footprint

Which one to choose?
Amazon-like: smaller footprint; well-connected (Multi-PoP)
Akamai-like: large footprint; better coverage (CDN)

The author studied the question using 20,000 IP addresses of Akamai hosts, and Akamai-like footprint was on par with Amazon-like footprint.

Note: this comparison does not include location-dependent middleboxes

Design Space - Provider Footprint

Location Dependent Middleboxes
Optimize for both **latency** and **bandwidth**

Akamai: 20% sub-milliseconds latency; 90% less than 5ms latency
Amazon: 30% less than 5ms latency (for US Clients)

Design Space - Provider Footprint

Location Dependent Middleboxes
Optimize for both **latency** and **bandwidth**

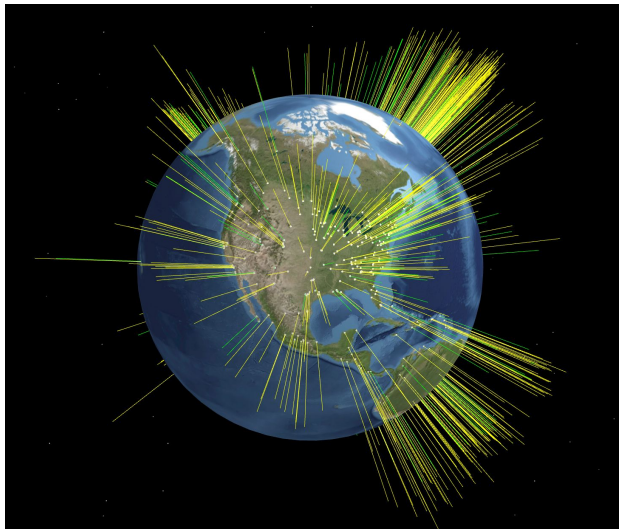
Akamai: 20% sub-milliseconds latency; 90% less than 5ms latency
Amazon: 30% less than 5ms latency (for US Clients)

Discussion:
The paper is published (more than) a decade ago. Given current service / network of AWS (e.g. *Lambda*) / Akamai, would the result change?

Design Space - Provider Footprint

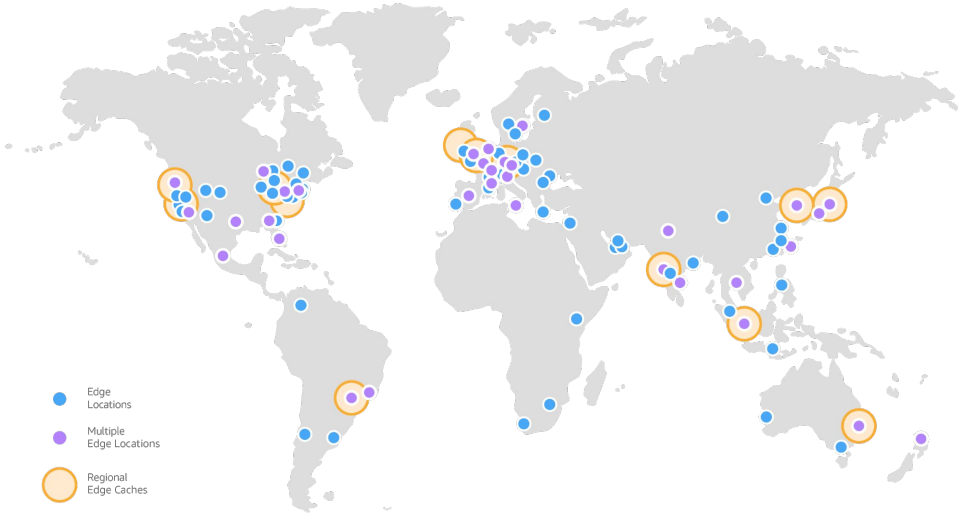
Location Dependent Middleboxes
Optimize for both **latency** and **bandwidth**

Akamai: 20% sub-milliseconds latency; 90% less than 5ms latency
Amazon: 30% less than 5ms latency (for US Clients)



Today's Akamai footprint

Source :http://wwwnui.akamai.com/globe/



Today's Amazon CloudFront footprint

Source: https://aws.amazon.com

Note: both pages are accessed on 02/23/2023

Design Space - Provider Footprint

Location Dependent Middleboxes
Optimize for both **latency** and **bandwidth**

Akamai: 20% sub-milliseconds latency; 90% less than 5ms latency
Amazon: 30% less than 5ms latency (for US Clients)

What about bandwidth?

APLOMB+: APLOMB w/ general-purpose traffic compression

Design Space - Options

Type of Middlebox	Enterprise Device	Cloud Footprint
IP Firewalls	Basic APLOMB	Multi-PoP
Application Firewalls	Basic APLOMB	Multi-PoP
VPN Gateways	Basic APLOMB	Multi-PoP
Load Balancers	Basic APLOMB	Multi-PoP
IDS/IPS	Basic APLOMB	Multi-PoP
WAN optimizers	APLOMB+	CDN
Proxies	APLOMB+	CDN

Table 2: Complexity of design and cloud footprint required to outsource different types of middleboxes.

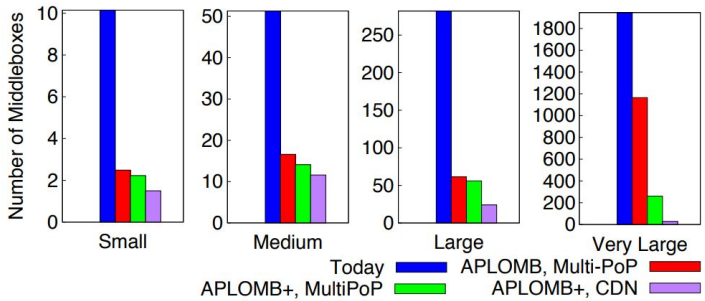


Figure 9: Average number of middleboxes remaining in enterprise under different outsourcing options.

APLOMB

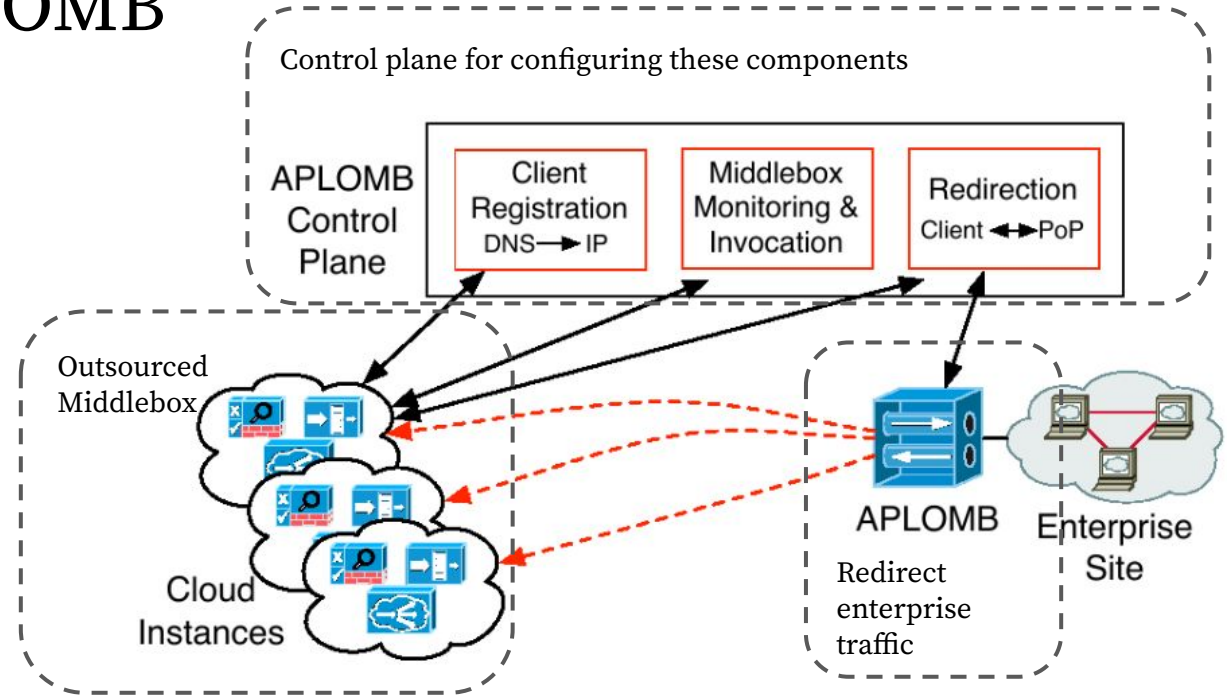


Figure 10: Architectural components of APLOMB.

APLOMB

To use APLOMB:

1. Configuration: Admins give service providers their address allocation
2. Registration: Associates address block in private address space to
3. APLOMB Gateway: Persistent tunnel to cloud PoPs; direct outbound traffic to appropriate cloud PoPs

APLOMB

To use APLOMB:

1. Configuration: Admins provide service providers their address allocation
2. Registration: Associates address block in private address space to
   a. Protected service: inter-site traffic via Internet-destined connections (no public IP is allocated)
   b. DNS service: Redirect incoming traffic (clients) to an appropriate cloud PoP via DNS redirection.
   c. Legacy service: enure backward-compatibility (e.g. services require fixed IP address)
3. APLOMB Gateway: Persistent tunnel to cloud PoPs; direct outbound traffic to appropriate cloud PoPs

APLOMB

To use APLOMB:

1. Configuration: Admins provide service providers their address allocation
2. Registration: Associates address block in private address space to
3. APLOMB Gateway: Persistent tunnel to cloud PoPs; direct outbound traffic to appropriate cloud PoPs

The Cloud

- Tunnel Endpoints: handle traffic to enterprise
- Middlebox Instances: process client traffic
- NAT (Network Address Translation) Devices: maintain IP-IP (for DNS and Legacy) / IP-Port mappings (for protected service)
- Policy switching logic: direct traffic between the above parts

Control Plane

Goal: keep APLOMB devices as simple and stateless as possible

- To optimize PoP selection: gather RTT from PoP to prefix on the Internet
- Adaptive scaling: gather utilization statistics

Control Plane

Goal: keep APLOMB devices as simple and stateless as possible

- To optimize PoP selection: gather RTT from PoP to prefix on the Internet
- Adaptive scaling: gather utilization statistics

- **Redirection Optimization**: pick the best PoP (minimizing the sum of two-side latency)
- **Policy Configuration**: Pipelining middleboxes (e.g. Firewall → IDS for incoming traffic)
- **Middlebox Scaling**: Dynamically allocate / deallocate middlebox instances based on the gathered utilization data

Sum up: the Implementation

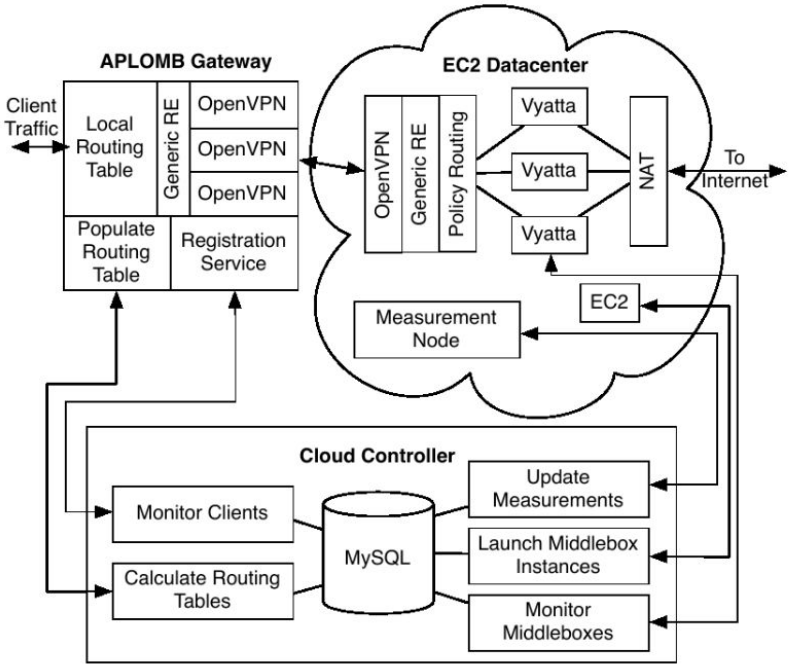


Figure 11: Software architecture of APLOMB.

Evaluation

1. **Application Performance**
   Benchmark for common applications.

2. **Scaling and Failover**
   The capacity to adapt various network load.

3. **Deployment Case Study**
   The feasibility of outsourcing middlebox functionality in an enterprise.

Application Performance

1. **Latency - HTTP Page Loads**

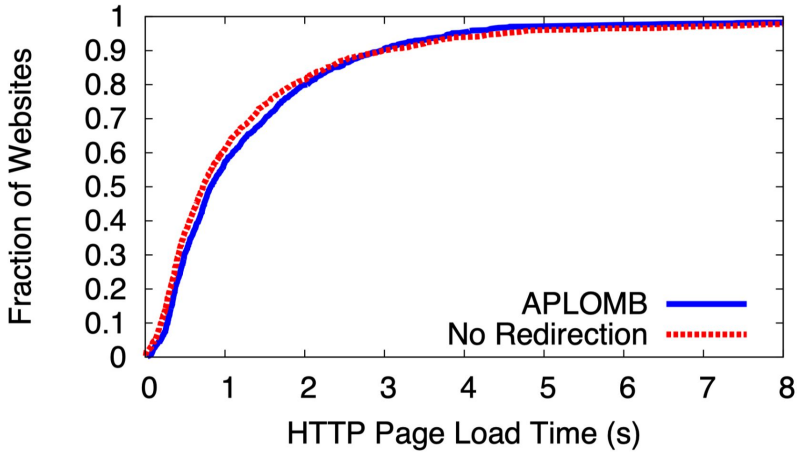


Figure 12: CDF of HTTP Page Load times for Alexa top 1,000 sites with and without APLOMB.

Application Performance

1. **Latency - HTTP Page Loads**

2. **Throughput - BitTorrent**
   Speed decreased ~5%

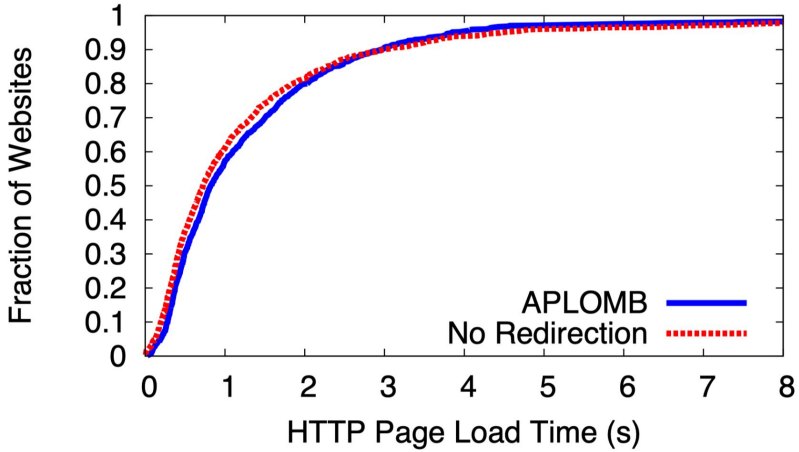


Figure 12: CDF of HTTP Page Load times for Alexa top 1,000 sites with and without APLOMB.

Application Performance

1. **Latency - HTTP Page Loads**

2. **Throughput - BitTorrent**
   Speed decreased ~5%

3. **Jitter - Voice over IP**

	Before(i/o)	After(i/o)
Residential network	2.3 ms/1.03 ms	2.49 ms/2.46 ms
Public WiFi hotspot	4.41 ms/4.04 ms	13.21 ms/14.49 ms

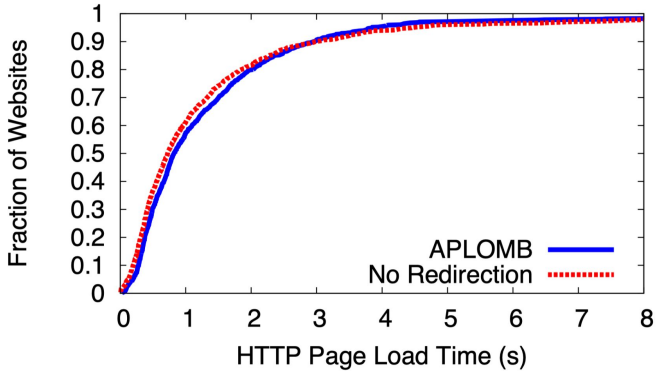


Figure 12: CDF of HTTP Page Load times for Alexa top 1,000 sites with and without APLOMB.

Scaling and Failover

1. **Dynamic Scaling**
   Streaming a video, repeatedly requesting large files over HTTP, and downloading several large files via BitTorrent over a 10-minute period.

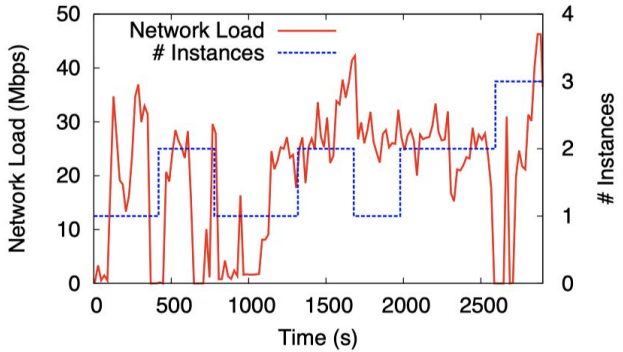


Figure 13: Network load (Y_1) and number of software middlebox instances (Y_2) under load. Experiment used low-capacity instances to highlight scaling dynamics.

Scaling and Failover

1. **Dynamic Scaling**
   Streaming a video, repeatedly requesting large files over HTTP, and downloading several large files via BitTorrent over a 10-minute period.

2. **Failure Handling**
   APLOMB checks for reachability between itself and individual middlebox instances every second.

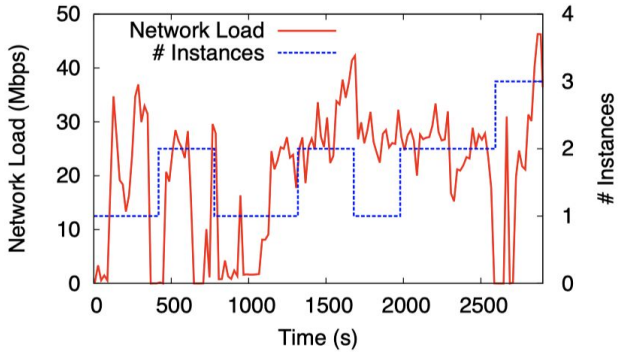


Figure 13: Network load (Y_1) and number of software middlebox instances (Y_2) under load. Experiment used low-capacity instances to highlight scaling dynamics.

Deployment Case Study

Goal: *Outsourcing as many middleboxes as possible and reducing enterprise costs, all the while without increasing bandwidth utilization or latency.*

1. **Middlebox Outsourced**
   ~60% of the middleboxes can be outsourced with APLOMB+.

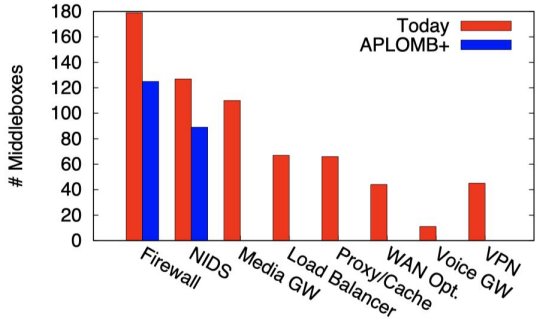


Figure 14: Number of middleboxes in the enterprise with and without APLOMB+. The enterprise has an atypical number of 'internal' firewalls and NIDS.

Deployment Case Study

Goal: *Outsourcing as many middleboxes as possible and reducing enterprise costs, all the while without increasing bandwidth utilization or latency.*

1. **Middlebox Outsourced**
   ~60% of the middleboxes can be outsourced with APLOMB+.

2. **Cost Reduction**
   Over **2×** peak-to-mean ratio.

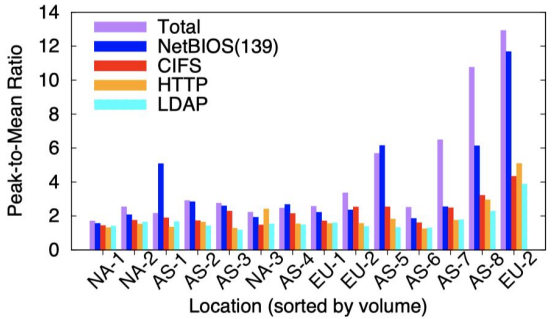


Figure 15: Ratio of peak traffic volume to average traffic volume, divided by protocol.

Deployment Case Study

Goal: *Outsourcing as many middleboxes as possible and reducing enterprise costs, all the while without increasing bandwidth utilization or latency.*

1. **Middlebox Outsourced**
   ~60% of the middleboxes can be outsourced with APLOMB+.

2. **Cost Reduction**
   Over 2× peak-to-mean ratio.

3. **Latency**
   More than 60% of inter-site pairs remains almost same latency.
   In expectation, a packet experiences only **1.13 ms** of inflation.

Deployment Case Study

Goal: *Outsourcing as many middleboxes as possible and reducing enterprise costs, all the while without increasing bandwidth utilization or latency.*

1. **Middlebox Outsourced**
   ~60% of the middleboxes can be outsourced with APLOMB+.

2. **Cost Reduction**
   Over 2× peak-to-mean ratio.

3. **Latency**
   More than 60% of inter-site pairs remains almost same latency.
   In expectation, a packet experiences only **1.13 ms** of inflation.

4. **Bandwidth**
   With generic RE, APLOMB+ reduces bandwidth utilization by **28%**.

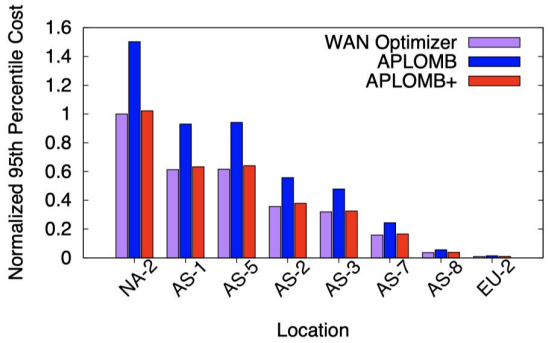


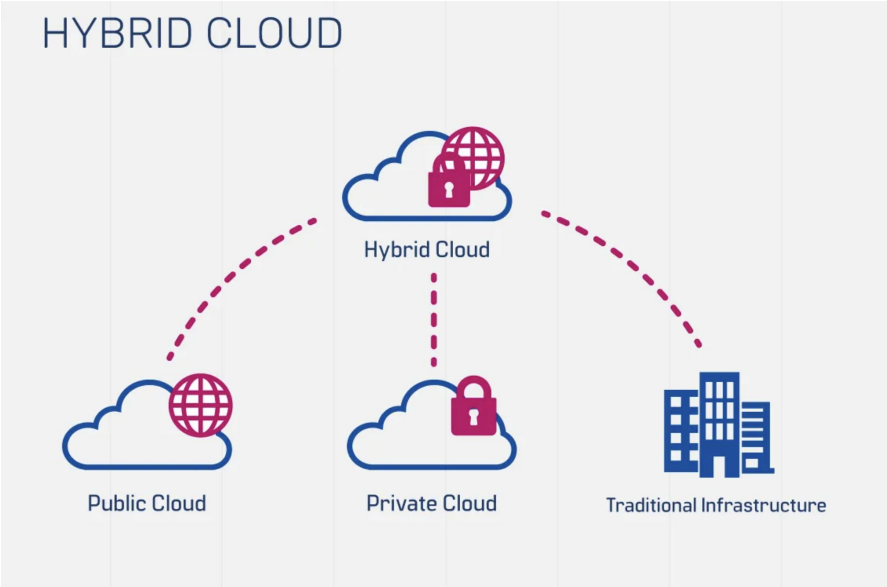
Figure 16: 95^{th} percentile bandwidth without APLOMB, with APLOMB, and with APLOMB+.

Discussion

- IT Outsourcing and Hybrid Clouds
- Bandwidth Costs
- Security Challenges

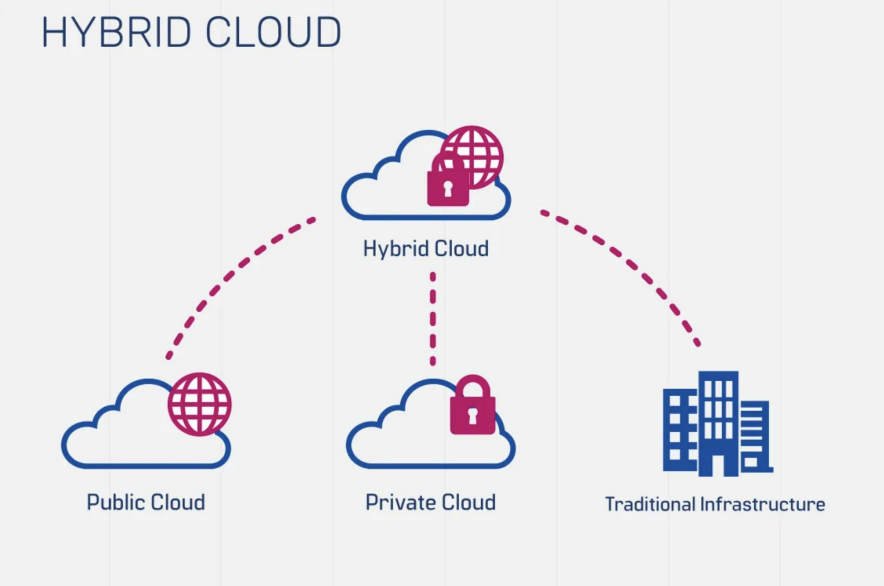
IT Outsourcing and Hybrid Clouds

- Some enterprises prefer to keep some local infrastructure due to security and performance concerns.
- User-facing devices such as laptops and smartphones will always remain within the enterprise.



IT Outsourcing and Hybrid Clouds

- APLOMB allows administrators to consolidate middleboxes in only one deployment setting.
- This evade the middlebox-related complexity in the hybrid model.



Bandwidth Costs

- APLOMB may increase bandwidth costs due to current cloud business models.

Bandwidth Costs

- APLOMB may increase bandwidth costs due to current cloud business models.
- Tunneling traffic to a cloud provider necessitates paying for bandwidth **twice**
    1. Enterprise network's access link
    2. At the cloud provider

Bandwidth Costs

- APLOMB may increase bandwidth costs due to current cloud business models.
- Tunneling traffic to a cloud provider necessitates paying for bandwidth twice
    1. Enterprise network's access link
    2. At the cloud provider

Does that mean APLOMB will double bandwidth costs for an enterprise?

Bandwidth Costs

- APLOMB may increase bandwidth costs due to current cloud business models.
- Tunneling traffic to a cloud provider necessitates paying for bandwidth twice
    1. **Enterprise network's access link**
    2. At the cloud provider

Does that mean APLOMB will double bandwidth costs for an enterprise?

- No. Redundancy elimination and compression can reduce bandwidth demands at the enterprise access link by roughly **30%**.

Bandwidth Costs

- APLOMB may increase bandwidth costs due to current cloud business models.
- Tunneling traffic to a cloud provider necessitates paying for bandwidth twice
  1. **Enterprise network's access link**
  2. At the cloud provider

Does that mean APLOMB will double bandwidth costs for an enterprise?

- No. Redundancy elimination and compression can reduce bandwidth demands at the enterprise access link by roughly **30%**.
- Redirection through a cloud PoP => Low capacity & Less expensive access link

Bandwidth Costs

- Tunneling traffic to a cloud provider necessitates paying for bandwidth twice
  1. Enterprise network's access link
  2. **At the cloud provider**
     - A dedicated APLOMB service provider could take advantage of wholesale bandwidth, which is priced by transfer rate, offering substantially lower prices than current cloud pricing strategies.

Bandwidth Costs

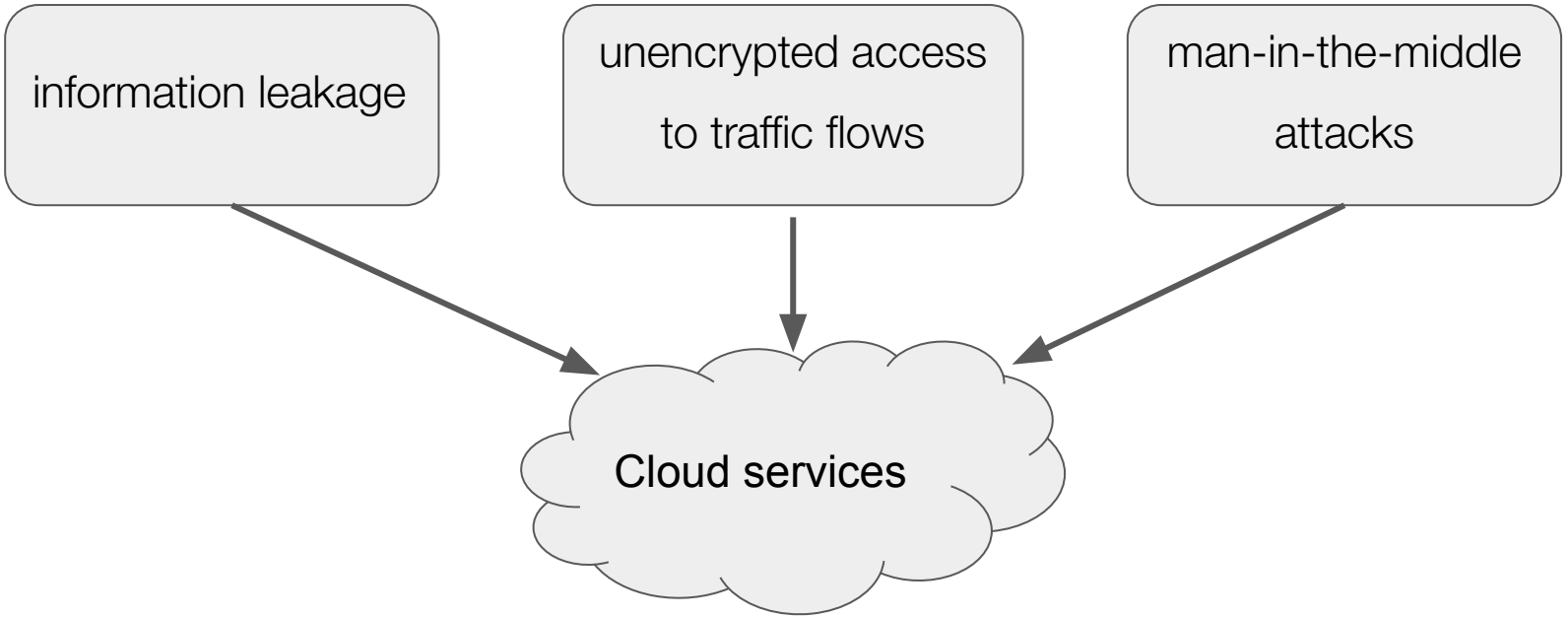
- Tunneling traffic to a cloud provider necessitates paying for bandwidth twice
  1. Enterprise network's access link
  2. **At the cloud provider**

Pricing Model	Total Cost	$/GB	$/Mbps
Standard EC2	30003.20	0.0586	17.58
Amazon DirectConnect	11882.50	0.0232	6.96
Wholesale Bandwidth	6826.70	0.0133	4.00

Table 3: Cost comparison of different cloud bandwidth pricing models given an enterprise with a monthly transfer volume of 500TB (an overestimate as compared to the very large enterprise in our study); assumes conversion rate of 1Mbps of sustained transfer equals 300GB over the course of a month.

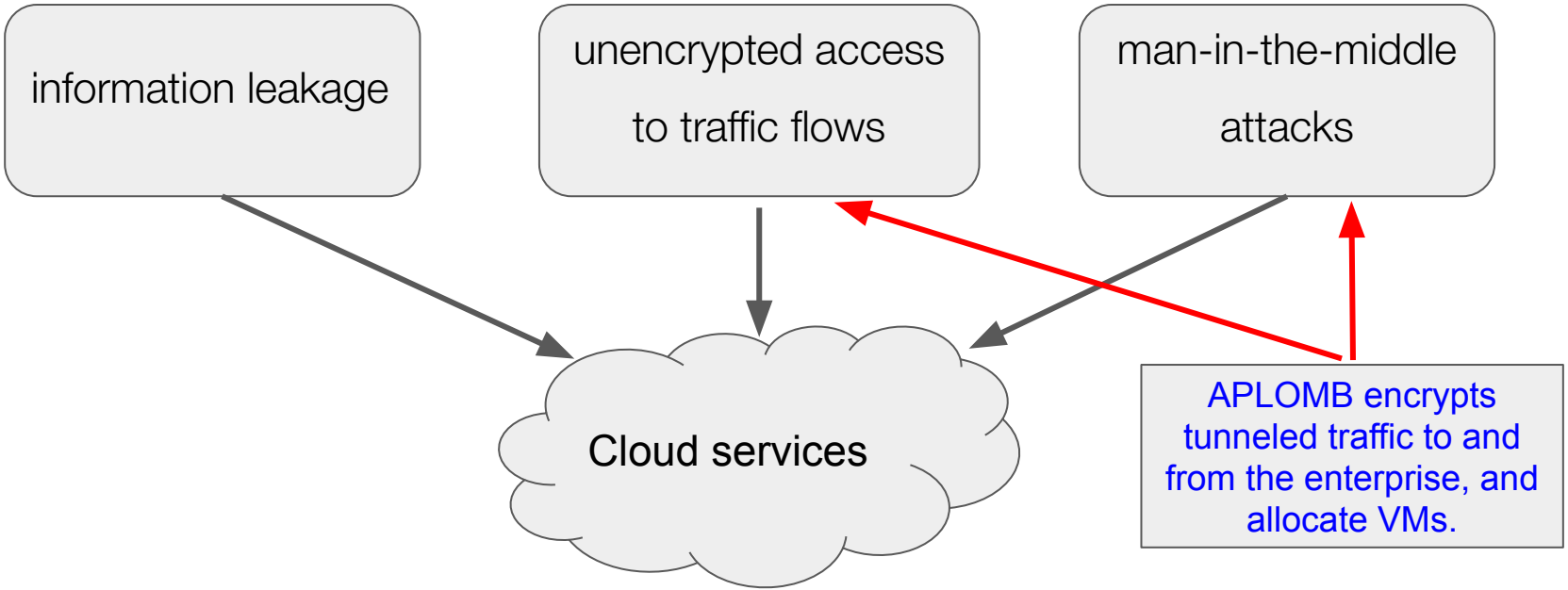
Security Challenges

Adopting APLOMB raises security concerns similar to those of cloud computing.



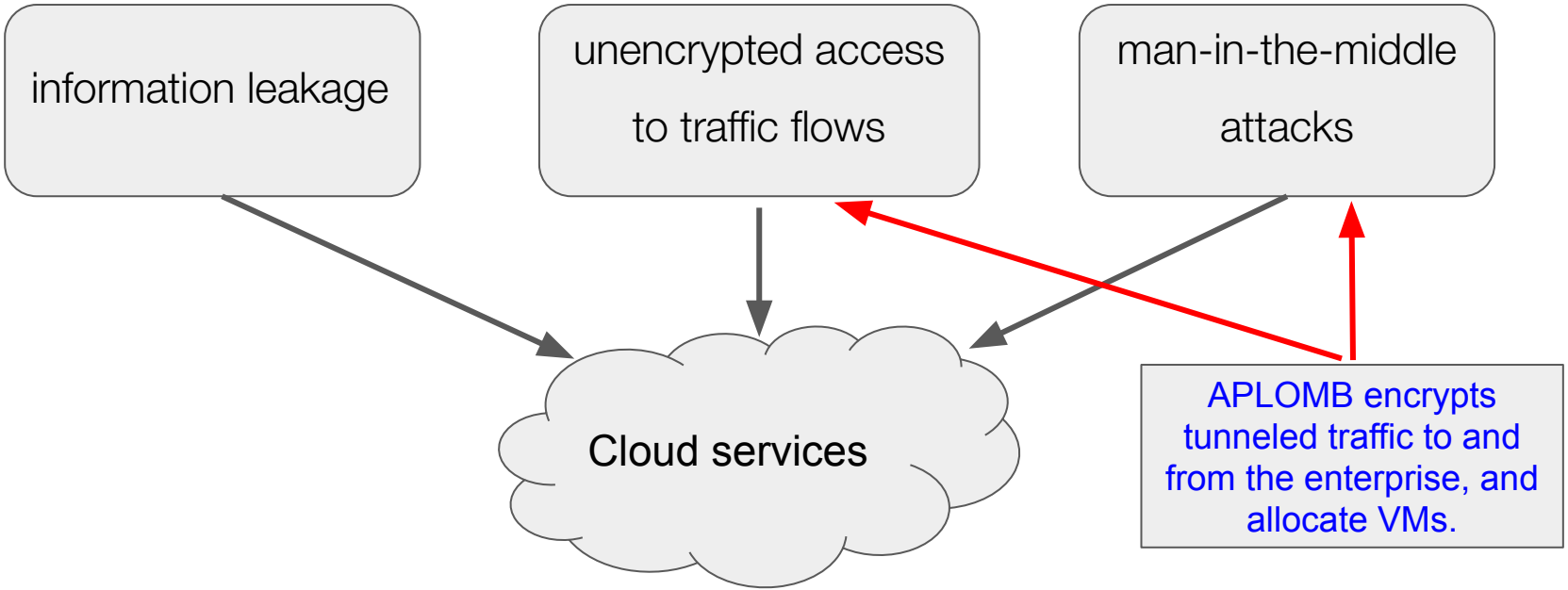
Security Challenges

Adopting APLOMB raises security concerns similar to those of cloud computing.



Security Challenges

Some security policies may restrict companies from using cloud-based services (APLOMB)



Related work

- Cloud Computing
- Middlebox Management
- Redirection Services
- Cloud Networking

Related work

Cloud computing

- The motivation for APLOMB parallels traditional arguments in favor of cloud computing. [1]
- APLOMB also adapts techniques from traditional cloud solutions
  - utilization monitoring and dynamic scaling. [2]
- DNS-based redirection to datacenters. [3]

[1] M. Armbrust et al. A view of cloud computing. Commun. ACM, April 2010.

[2] Rightscale Cloud management. http://www.rightscale.com/.

[3] A. Su, D. Choffnes, A. Kuzmanovic, and F. Bustamante. Drafting behind Akamai (Travelocity-based detouring). In SIGCOMM, 2006.

Related work

Middlebox Management

- Many works have tackled middlebox management challenges within the enterprise.
  - the policy-routing switch [1]
  - the management plane [2]
  - consolidated appliance [3]
- ETTM proposes removing middleboxes from the enterprise network. [4]
  - Pushing middlebox processing to enterprise end hosts.
  - Retains the problem of middlebox management in the enterprise

[1] D. A. Joseph, A. Tavakoli, and I. Stoica. A policy-aware switching layer for data centers. In SIGCOMM, 2008.
[2] H. Ballani and P. Francis. CONMan: a step towards network manageability. In SIGCOMM, 2007.
[3] V. Sekar, S. Ratnasamy, M. K. Reiter, N. Egi, and G. Shi. The middlebox manifesto: enabling innovation in middlebox deployment. In HotNets, 2011.
[4] C. Dixon, H. Uppal, V. Brajkovic, D. Brandon, T. Anderson, and A. Krishnamurthy. ETTM: a scalable fault tolerant network manager. In NSDI, 2011.

Related work

Middlebox Management

- Sekar et al propose a consolidated middlebox architecture [1]
  - Reduce the workload related to managing middleboxes.
  - But still not removing middleboxes from the enterprise network entirely.

[1] V. Sekar, S. Ratnasamy, M. K. Reiter, N. Egi, and G. Shi. The middlebox manifesto: enabling innovation in middlebox deployment. In HotNets, 2011

Related work

Redirection Services

- Traffic redirection infrastructures have been explored to improve Internet or overlay routing architectures.
  - APLOMB's goal is to enable middlebox processing in the cloud.
- RON showed how routing via an intermediary might improve latency. [1]
  - APLOMB reports similar findings using cloud PoPs as intermediaries.

[1] D. Andersen, H. Balakrishnan, F. Kaashoek, and R. Morris. Resilient overlay networks. In SOSP, 2001.

Related work

Redirection Services

- Walfish et al. propose a clean-slate architecture, DOA, by which end hosts explicitly address middleboxes. [1]
- Gibb et al. develop a service model for middleboxes that focuses on service aware routers that redirect traffic to middleboxes that can be in the local network or Internet. [2]

[1] M. Walfish, J. Stribling, M. Krohn, H. Balakrishnan, R. Morris, and S. Shenker. Middleboxes no longer considered harmful. In OSDI, 2004.
[2] G. Gibb, H. Zeng, and N. McKeown. Outsourcing network functionality. In HotSDN, 2012

Related work

Cloud Networking

- Using virtual middlebox appliances reduces the physical hardware cost of middlebox ownership.
- But cannot match the performance of hardware solutions
- Cloud-based offerings for specific middlebox services:
  - protocol acceleration
  - intrusion detection
  - web security
- APLOMB is an extreme extrapolation of their services.