

upna

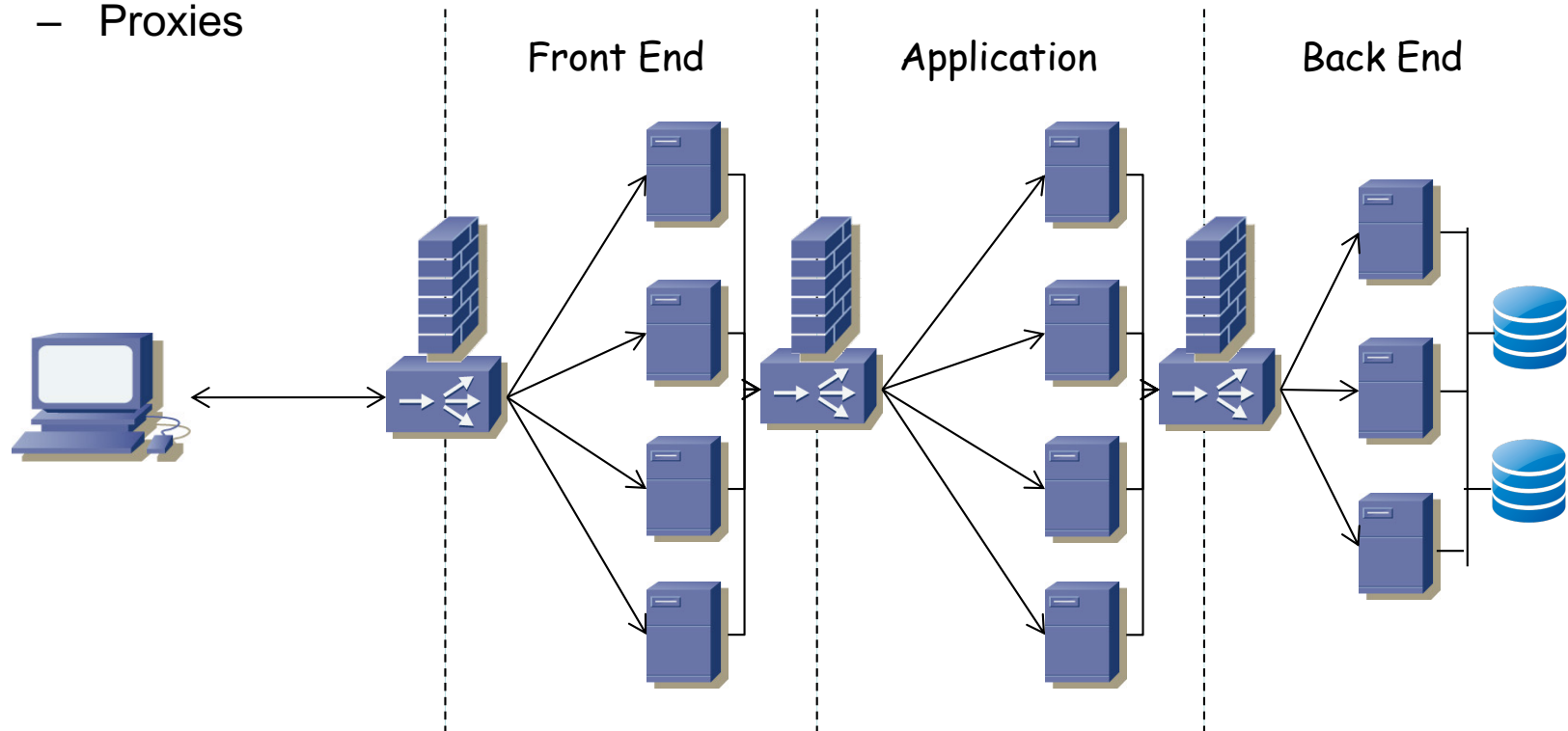
Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

Servicios de red

Servicios y multitier

- Hemos visto que es común la separación en capas del servicio
- Entre ellas nos podremos encontrar diferentes servicios:
 - Balanceadores de carga (*content switching*)
 - Firewalls
 - IDSs (Intrusion Detection Systems)
 - SSL Offloading
 - Caches
 - Proxies



upna

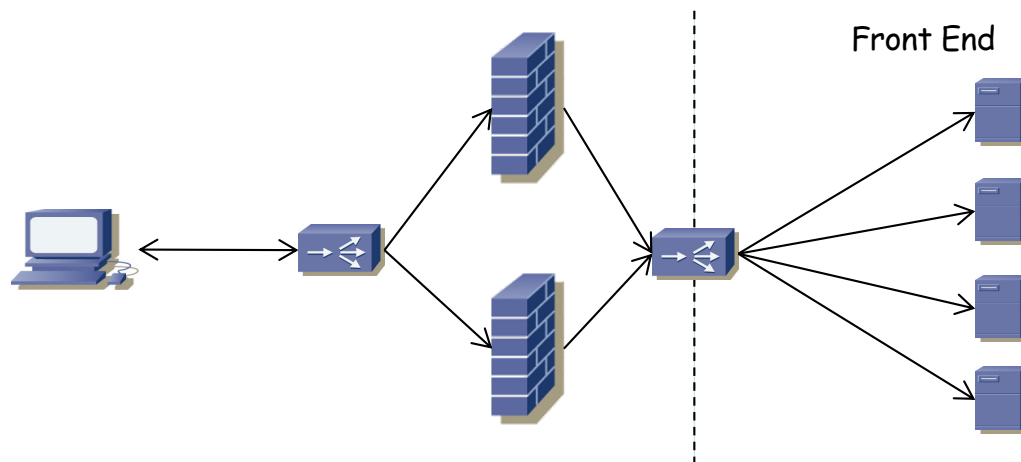
Universidad Pública de Navarra
Nafarroako Unibertsitate Publikoa

Redes de Nueva Generación
Área de Ingeniería Telemática

Otros servicios

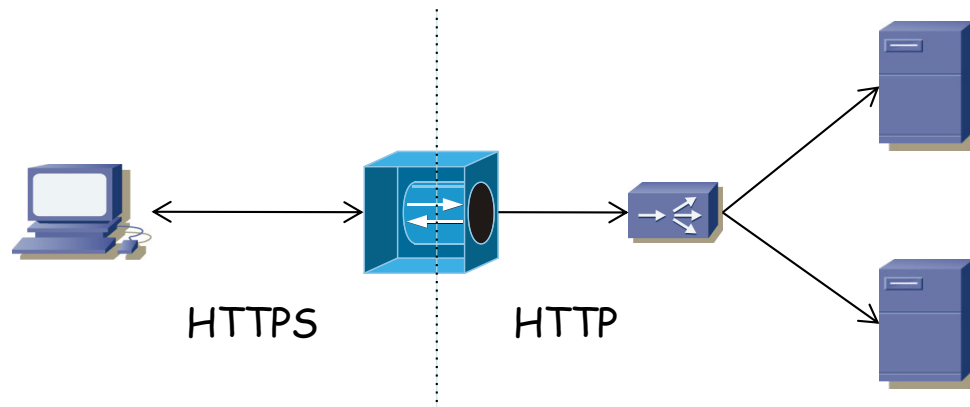
Firewalls e IDS

- Seguridad, seguridad, seguridad
- Reglas de filtrado para permitir el acceso solo a las direcciones IP y puertos de los servicios
- Inspección de contenido
- Pueden estar antes o después del balanceador
- Si no vale con uno se pueden poner varios balanceados (aumenta la complejidad)
- Ese balanceador podría ser el mismo que hacia los servidores (varias direcciones IP virtuales o instancias virtuales)



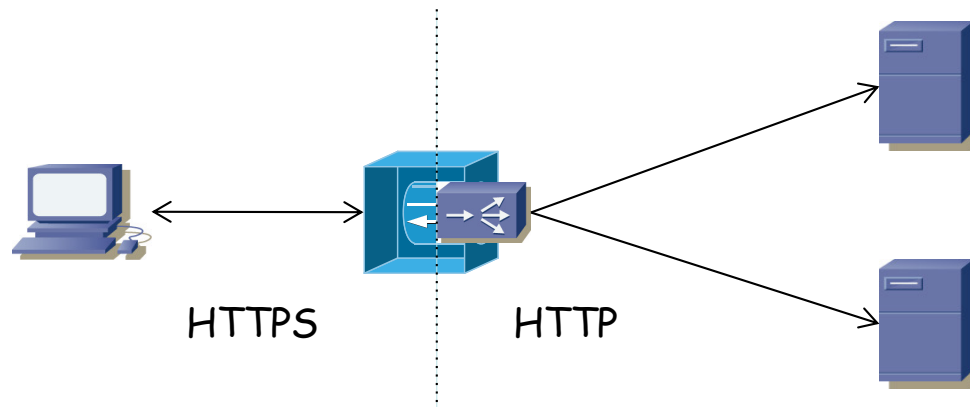
SSL offloading

- Portales web seguros
- También otros servicios sobre un túnel SSL
- SSL tiene un coste computacional considerable (¿hardware?)
- Este equipo termina la sesión SSL con el usuario e inicia una conexión sin SSL con el servidor
- El equipo puede disponer de hardware especializado para SSL
- También podríamos poner varios y balancearlos
- (...)



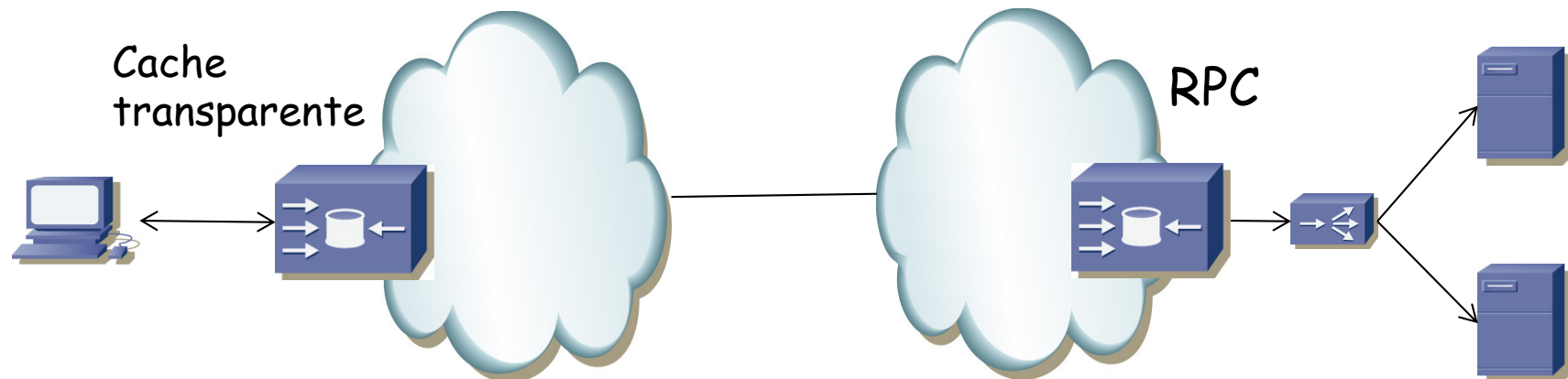
SSL offloading

- Portales web seguros
- También otros servicios sobre un túnel SSL
- SSL tiene un coste computacional considerable (¿hardware?)
- Este equipo termina la sesión SSL con el usuario e inicia una conexión sin SSL con el servidor
- El equipo puede disponer de hardware especializado para SSL
- También podríamos poner varios y balancearlos
- Es común que el balanceador integre esta funcionalidad



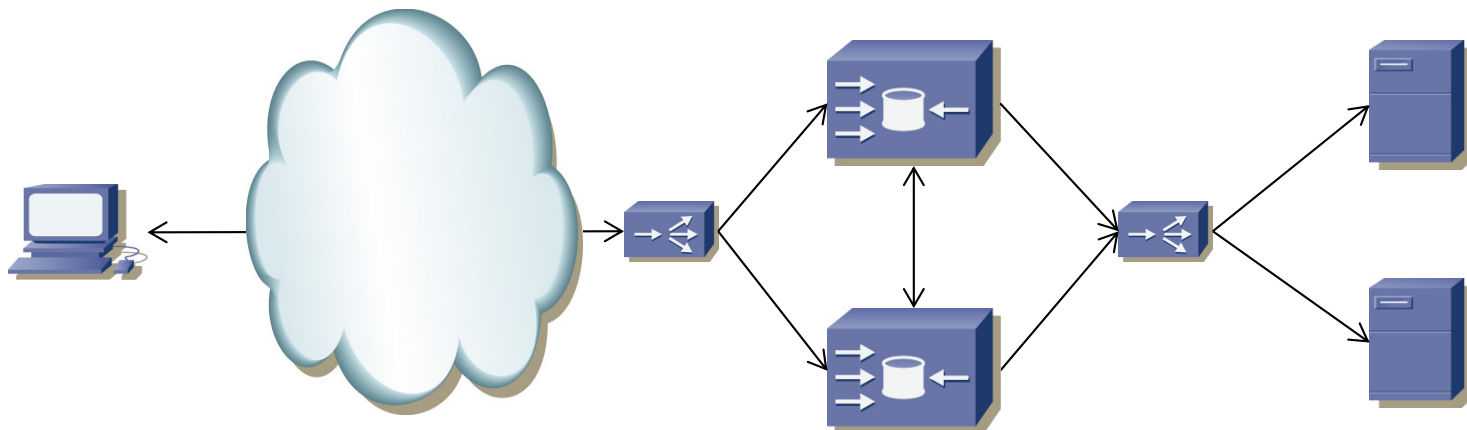
Cache

- Puede ser cercana a los servidores, a los clientes o a ambos
- Cercanas al servidor
 - Se habla de “*reverse proxy cache*” (RPC)
 - Reduce carga sobre los servidores
- Cercanas al cliente
 - Se habla de “*transparent caching*”
 - Reducen carga sobre el enlace a Internet
 - Reducen tiempos de respuesta por cercanía (menor RTT)



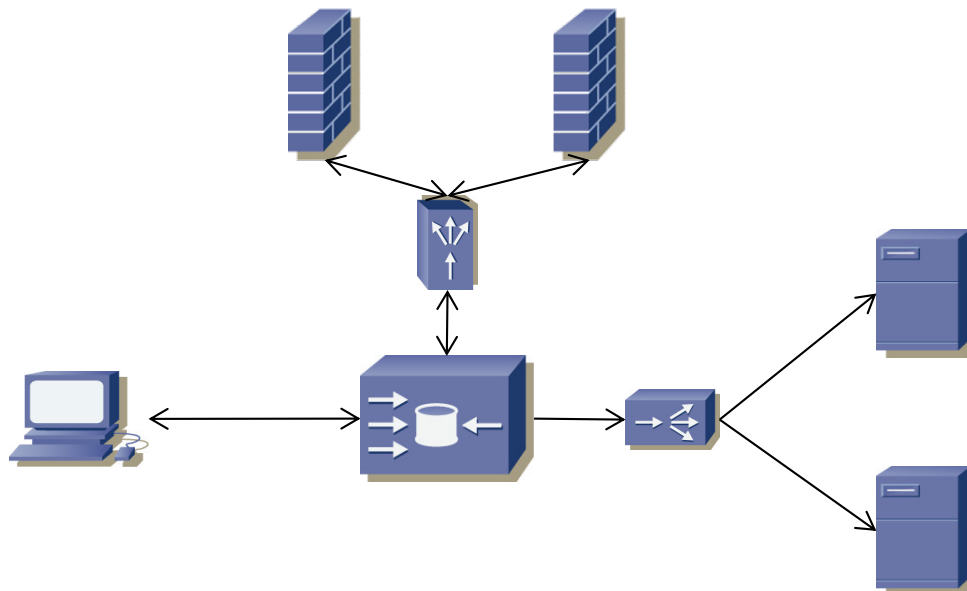
Cache

- La cache podría implementarse con varias caches balanceadas
 - Aumenta la capacidad (CPU) de la cache
 - Busca maximizar el *cache hit ratio* y así reducir peticiones a servidores
 - Para ello el balanceador debería reenviar la petición a la cache con mayor probabilidad de contenerlo (en función del FQDN)
 - O las caches deben sincronizarse (*clustering*), pues si no acabarán haciendo peticiones repetidas



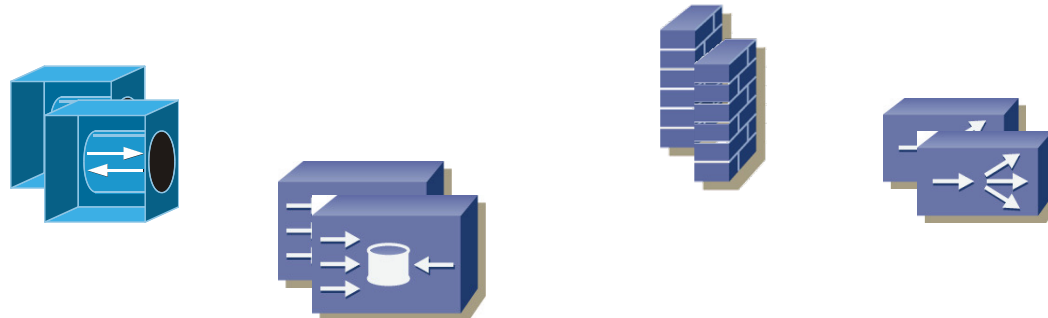
Cache

- Puede redirigir parte de la petición a un antivirus o filtro de contenido
- Se encargaría de verificar que se puede hacer esa petición o que el documento obtenido no es peligroso
- Protocolos específicos para pasar la petición o respuesta: ICAP = *Internet Content Adaptation Protocol* (RFC 3507)
- O a varios con balanceo de carga
- El balanceador puede ser el mismo equipo



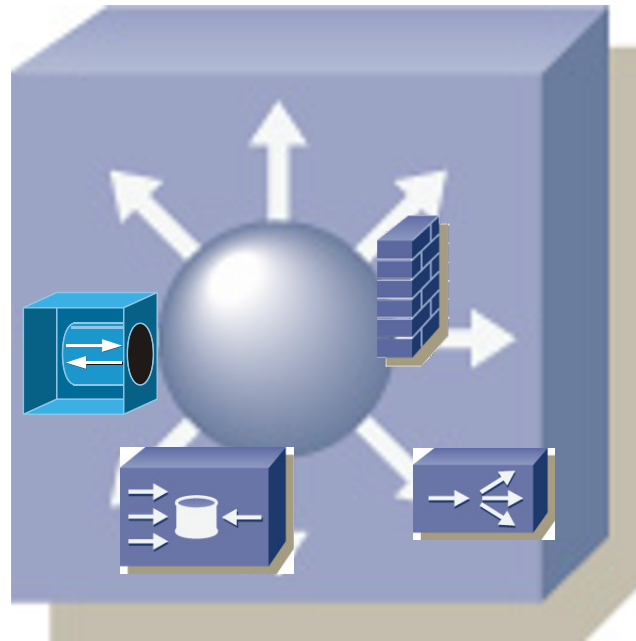
Servicios y redundancia

- Todos estos servicios se pueden dar desde equipos independientes
- Si no queremos un punto único de fallo debemos tenerlos replicados
- Según el tipo de servicio deberán coordinarse entre ellos para mantener el estado ante un fallo
- Por ejemplo un NAT para conocer las sesiones de mapeo que estaban establecidas



Servicios y redundancia

- Todos estos servicios se pueden dar desde equipos independientes
- Si no queremos un punto único de fallo debemos tenerlos replicados
- Según el tipo de servicio deberán coordinarse entre ellos para mantener el estado ante un fallo
- Por ejemplo un NAT para conocer las sesiones de mapeo que estaban establecidas
- También pueden ser módulos en un conmutador



VMs como Firewalls

- <https://www.paloaltonetworks.com/prisma/vm-series>
- <https://www.fortinet.com/products/private-cloud-security/fortigate-virtual-appliances#models-specs>
- <https://www.checkpoint.com/products/iaas-private-cloud-security/>

Secure innovation v
always-on, anytime se

VM-Series virtual firewalls help your organization safely transform its virtual business innovation and competitiveness.

[Read public cloud white paper](#)

Automate netw

VM-Series virtual firewa provisioning directly into lifecycle and CI/CD pipe demand scalability.

Improve netwo posture

VM-Series virtual firewa manage network securit premises, in private and branch locations.

FORTINET Support Training Resources Partners Corporate

SECURITY-DRIVEN NETWORKING ADAPTIVE CLOUD SECURITY AI-DRIVEN SECURITY OPERATIONS ZERO TRUST ACCESS

Overview **Models & Specs** Public Cloud Private Cloud Carrier VNF Resources

FortiGate-VM

FortiGate Virtual Next-generation Firewall Models and Specifications

FortiGate-VM next-generation firewall can be deployed as a virtual appliance in private and public cloud environments, either as a BYOL instance or provisioned on-demand via public cloud marketplaces.

[Download the brief](#) - Performance as a key attribute of Virtual Firewalls.

[Compare Products](#)

Supported Configurations

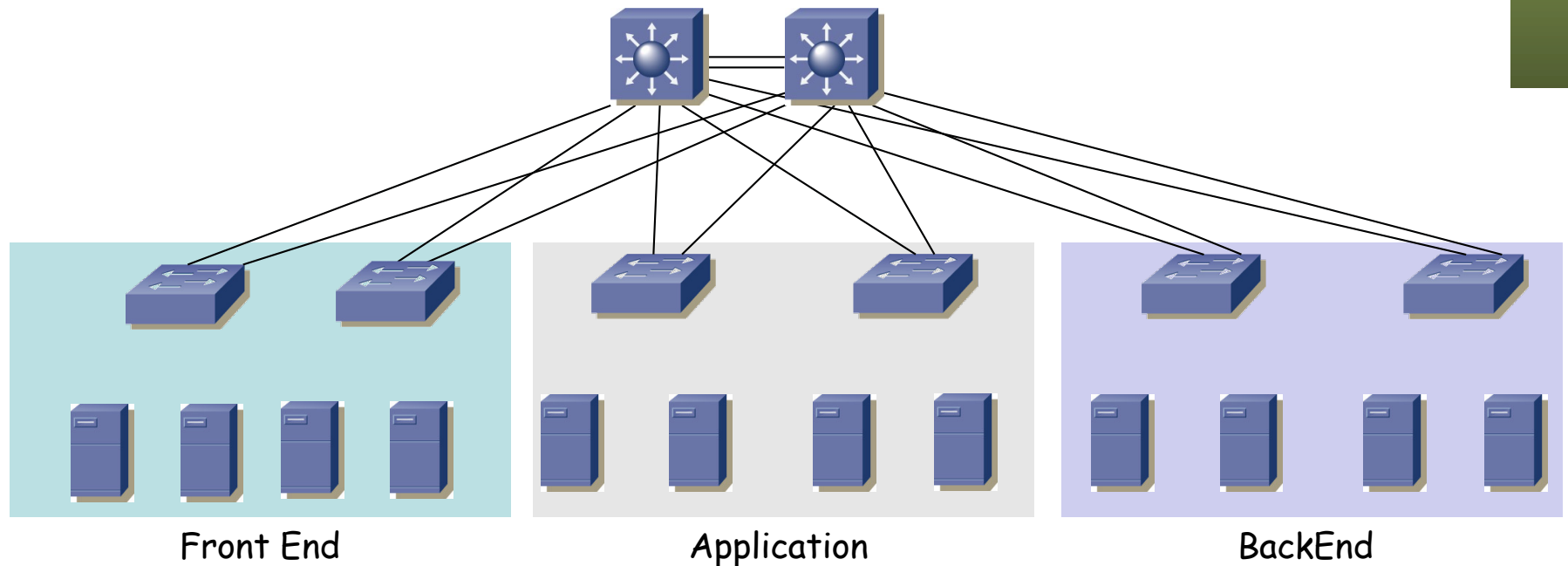
Private Cloud Platforms	vmware ESXi	Microsoft Hyper-V	KVM
Supported Solutions and Releases	VMware vSphere v5 or later	Windows Server 2012 R2 Windows Server 2016 Windows Server 2019	REHL 7, CentOS 7, Ubuntu Server 18.04
Supported Solutions and Releases	VMware vSphere v5.5/6.0/6.5/6.7: R80.30, R80.40 VMware vSphere v7: R80.40	Windows Server 2012 R2: R80.30, R80.40 Windows Server 2016: R80.30, R80.40 Windows Server 2019: R80.40	R80.30, R80.40
Private Cloud Platforms	CISCO NSX	vmware NSX	openstack
Supported Solutions and Releases	APIC Version 1.3/2.0/2.1/2.2/2.3/3.0/3.1	NSX-V 6.3.x-6.4.x NSX-T 2.3.x/2.4.x/2.5.x/3.0.x	Newton, Ocata, Pike
Supported Solutions and Releases	R80.30, R80.40	NSX-V: R80.10 NSX-T: R80.30	R80.30, R80.40

FortiGate-VM00	FortiGate-VM01, -VM01V
Throughput: 12 Gbps vCPU: 1x vCPU core, (up to) 2 GB RAM	Throughput: 12 Gbps vCPU: 1x vCPU core, (up to) 2 GB RAM
FortiGate-VM02, -VM02V	FortiGate-VM04, -VM04V
Throughput: 15 Gbps vCPU: 2x vCPU cores, (up to) 4 GB RAM	Throughput: 28 Gbps vCPU: 4x vCPU cores, (up to) 6 GB RAM
FortiGate-VM08, -VM08V	FortiGate-VM16, -VM16V
Throughput: 33 Gbps vCPU: 8x vCPU cores, (up to) 12 GB RAM	Throughput: 36 Gbps vCPU: 16x vCPU cores, (up to) 24 GB RAM

Ubicación de los servicios

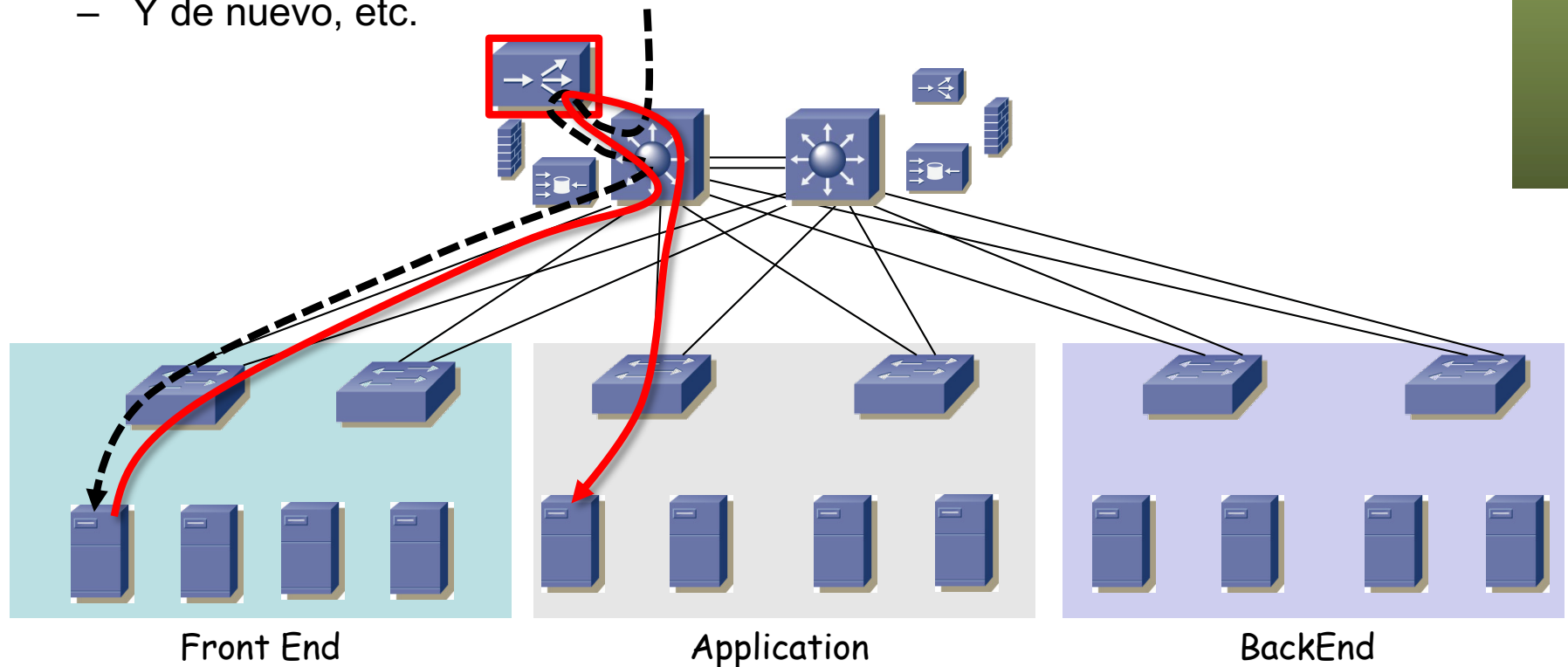
Servicios: ¿Dónde?

- Es común que los *tiers* estén en la capa de acceso
- Con un diseño colapsado los servicios estarían conectados a los conmutadores de agregación (...)



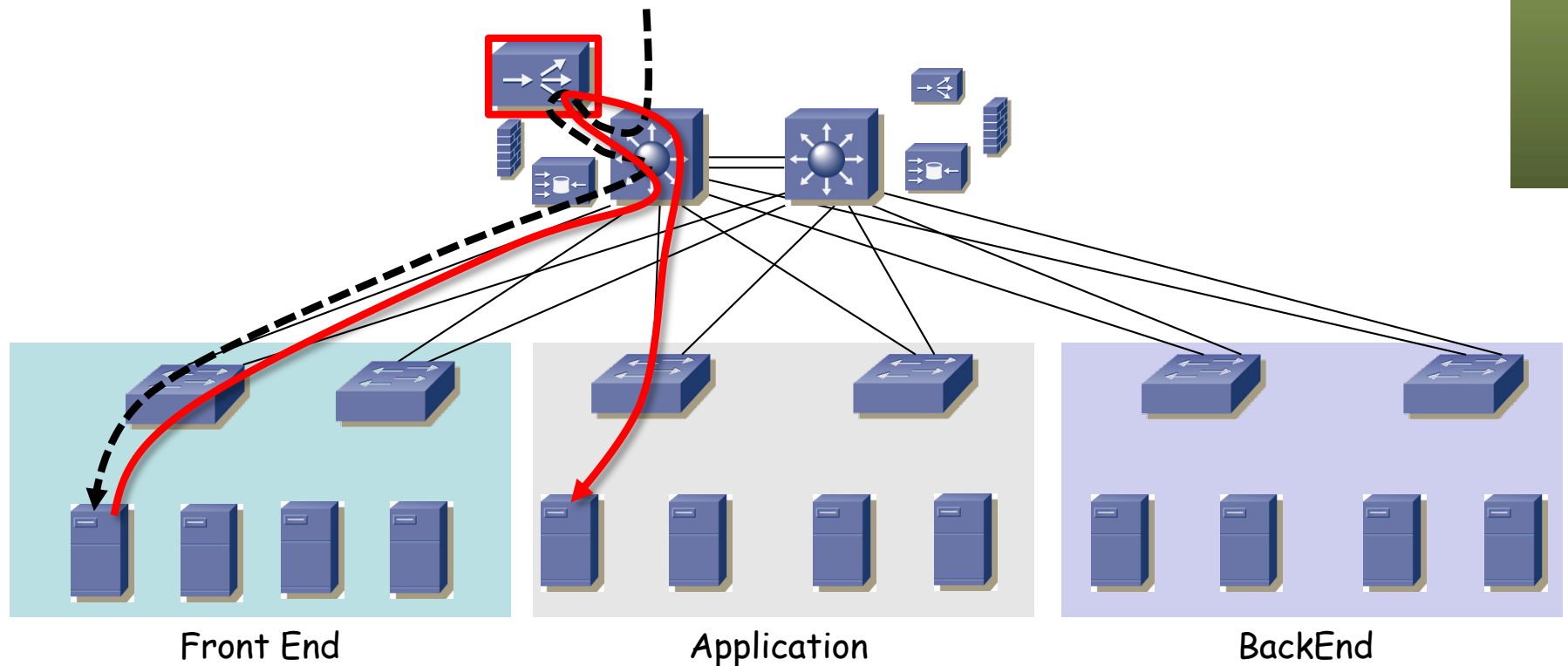
Servicios: ¿Dónde?

- Es común que los *tiers* estén en la capa de acceso
- Con un diseño colapsado los servicios estarían conectados a los conmutadores de agregación
- O pueden ser módulos en los conmutadores de agregación
- Pueden ser compartidos entre las diferentes capas
- Por ejemplo el mismo balanceador
 - Pasa por el balanceador de camino al *front end*
 - Y de nuevo, etc.



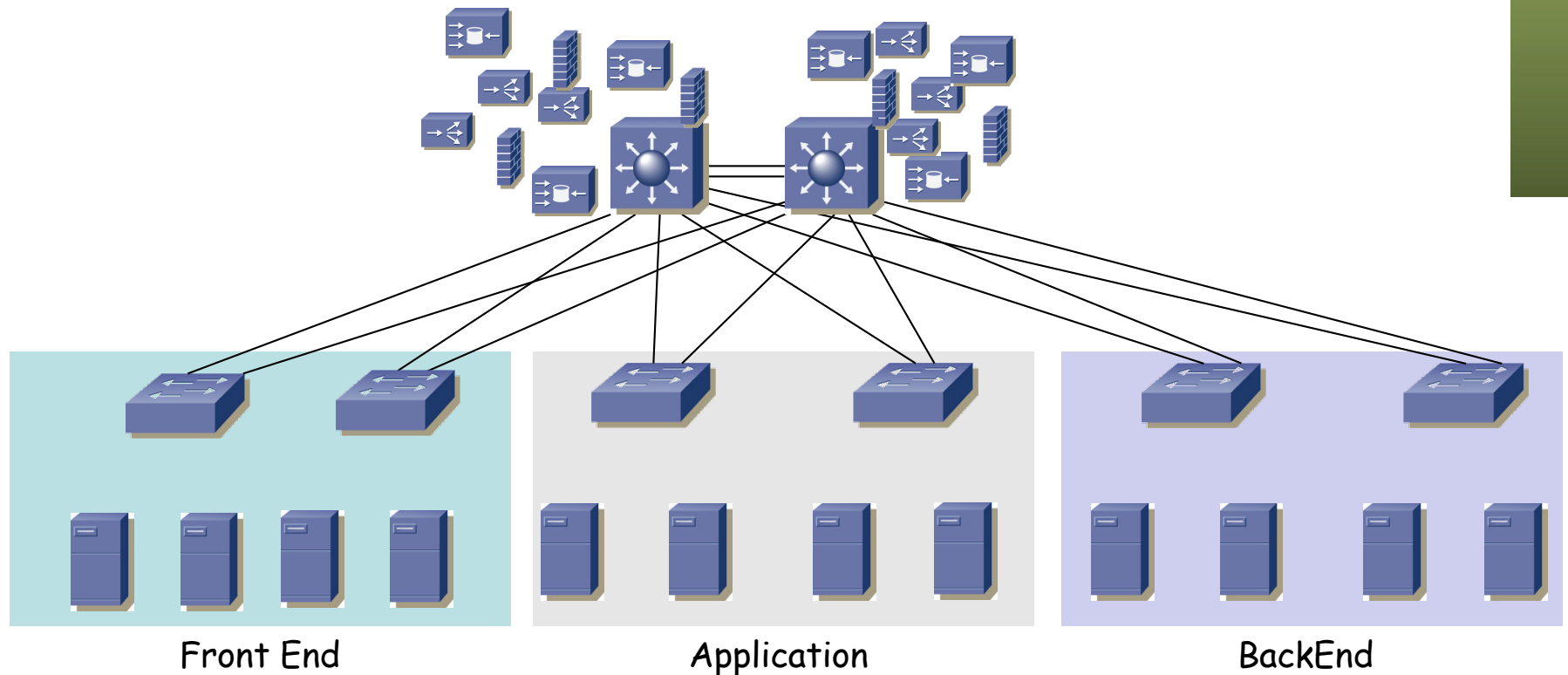
Servicios: ¿Dónde?

- Esto puede ser gracias a que tenga varios interfaces físicos, en las diferentes VLANs
- Porque emplee trunking en su(s) interfaz(-ces)
- Puede incluso dividirse en varios balanceadores “virtuales”
- Compartirlos reduce costes pero aumenta la complejidad y requiere mayor rendimiento de los mismos



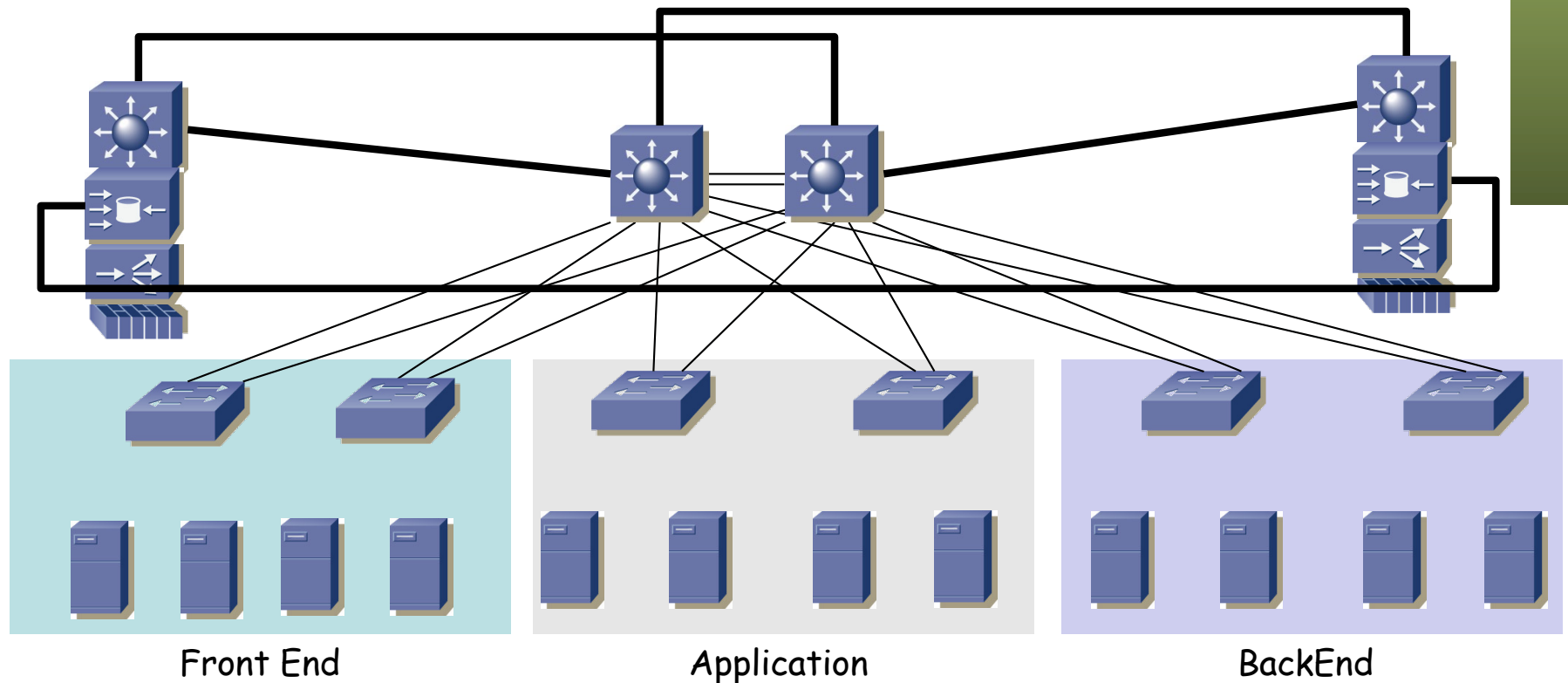
Servicios: ¿Dónde?

- Los equipos pueden ser demasiados para los slots de los conmutadores de agregación
- Demasiados para los puertos de los conmutadores de agregación
- (...)



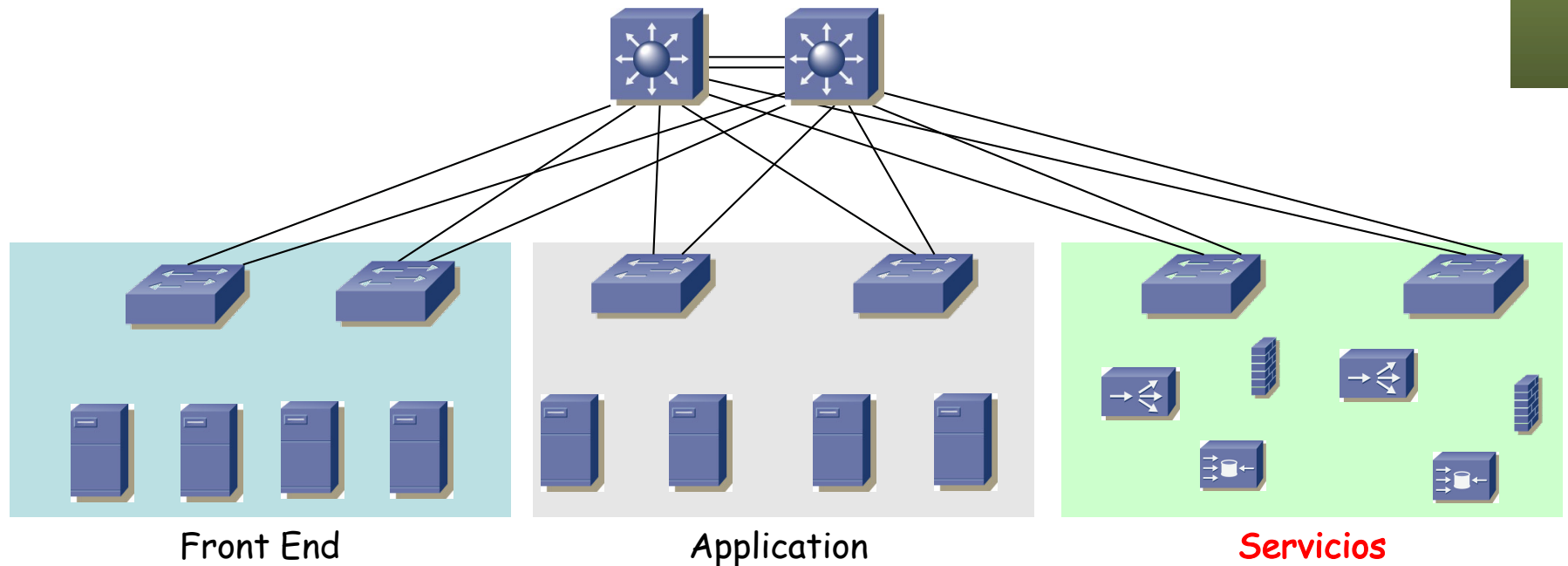
Servicios: ¿Dónde?

- Los equipos pueden ser demasiados para los slots de los conmutadores de agregación
- Demasiados para los puertos de los conmutadores de agregación
- Podemos sacarlos a sus propios conmutadores (*service switches*)
- (...)



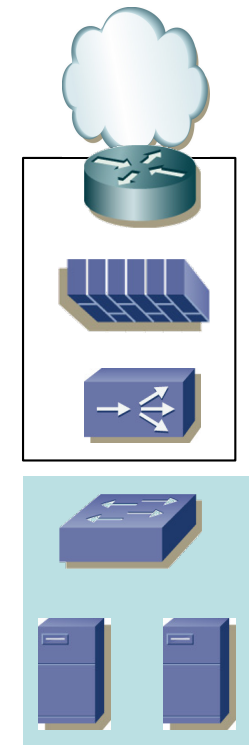
Servicios: ¿Dónde?

- Los equipos pueden ser demasiados para los slots de los conmutadores de agregación
- Demasiados para los puertos de los conmutadores de agregación
- Podemos sacarlos a sus propios conmutadores
- O sacarlos de la capa de agregación a su propia capa de acceso
- Especialmente necesario si son múltiples equipos balanceados



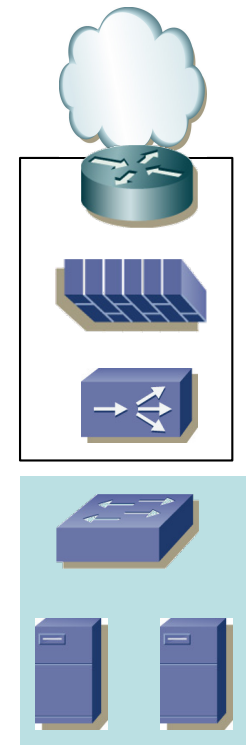
Orden de los servicios

- Recordemos que algunos de los servicios pueden ser módulos en un router/switch
- Tendremos diferentes formas de ordenarlos en el camino hacia los servidores
- Cada forma tendrá ventajas e inconvenientes
- (...)



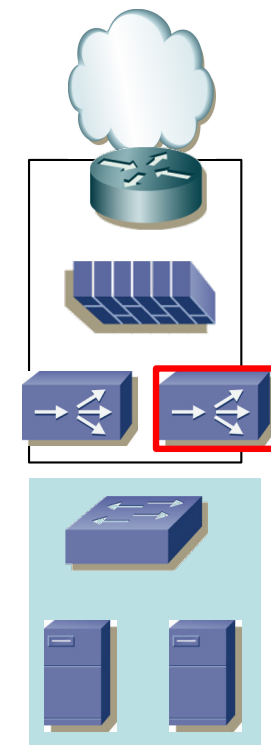
Router-Firewall-Balanceador

- Desde el núcleo, podemos encontrarnos primero con el router
- A continuación el firewall
- Finalmente el balanceador
- Si el balanceador se comporta como un puente entonces el router por defecto será el firewall
- Si el balanceador se comporta como un router se vuelve el router por defecto para los servidores



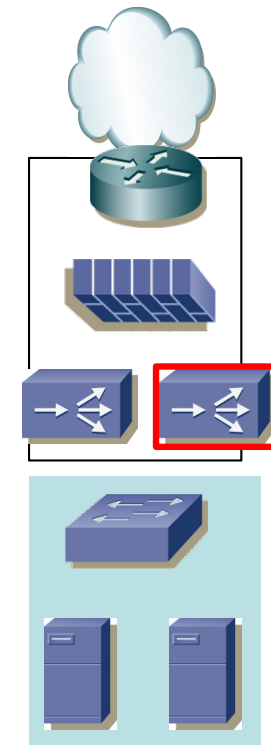
Router-Firewall-Balanceador

- Entre los elementos redundados estará el balanceador
- Si es el router por defecto puede emplear el FHRP
- Pueden configurarse en activo-pasivo o activo-activo
- Activo-pasivo (*active-standby*)
 - Uno de ellos hace todo el trabajo y si falla entra el otro
 - Mantener el estado sincronizado es sencillo (un solo sentido)
 - Es la alternativa más simple
- (...)



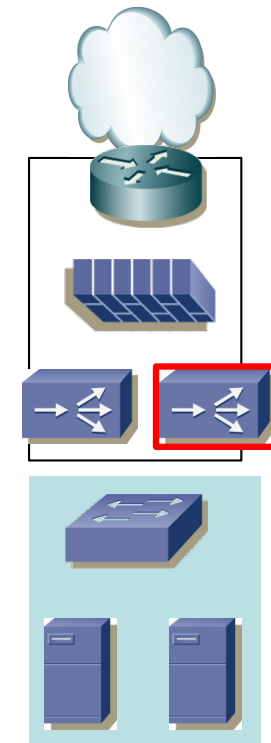
Router-Firewall-Balanceador

- Entre los elementos redundados estará el balanceador
- Si es el router por defecto puede emplear el FHRP
- Pueden configurarse en activo-pasivo o activo-activo
- Activo-pasivo (*active-standby*)
- Activo-activo (*active-active*) con reparto de VIPs
 - Las direcciones virtuales de los servicios se reparten
 - Cada dirección es empleada por un balanceador y el otro es el de respaldo
 - Es como emplear 2 grupos VRRP en la subred
 - Los servidores tendrán de router por defecto al balanceador que gestione como primario la dirección IP de su servicio
- (...)



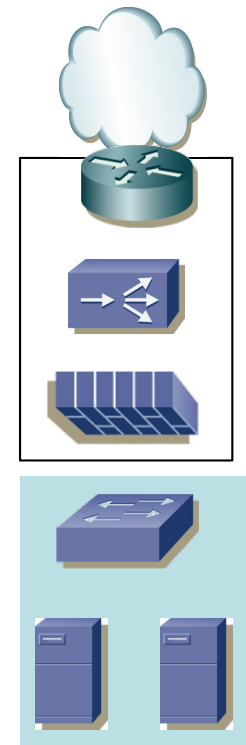
Router-Firewall-Balanceador

- Entre los elementos redundados estará el balanceador
- Si es el router por defecto puede emplear el FHRP
- Pueden configurarse en activo-pasivo o activo-activo
- Activo-pasivo (*active-standby*)
- Activo-activo (*active-active*) con reparto de VIPs
- Activo-activo con VIPs replicadas
 - Las direcciones IP de los servicios están activas en los dos
 - Hay que conseguir que el mismo cliente (toda su sesión) vaya siempre al mismo equipo
 - Esto es complejo pues los equipos *upstream* son conmutadores capa 2 y/o 3 que no entienden de sesiones
 - Normalmente eso requiere repartir a los clientes entre las dos instancias de la dirección IP virtual



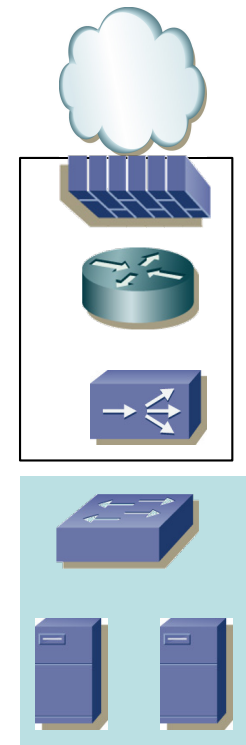
Router-Balanceador-Firewall

- En este caso tras el router está el balanceador y detrás el firewall
- El firewall debe permitir que el balanceador verifique el estado de los servidores (*health probes*)
- Esto implica configuración
- En esta configuración el router por defecto es el firewall
- Si el firewall actúa como router los *health probes* del balanceador deben ser enrutables



Firewall-Router-Balanceador

- En este caso la entrada es por el firewall
- Es probable que requiera funcionalidades extra de router como por ejemplo integrarse en el IGP
- Es más difícil securizar cada *tier* pues están todos al otro lado del firewall, enrutados sin pasar por el fw
- Según cómo opere el balanceador el router por defecto es él o el router



Firewall-Balanceador-Router

- El router como router por defecto para los servidores
- Eso permite usar funcionalidades habituales suyas como un FHRP, QoS, relay DHCP, etc.
- El balanceador no puede emplear una técnica que le requiera conectividad L2 con los servidores
- Los *health probes* que envíe el balanceador deben ser enrutables
- De nuevo pasar por el firewall entre cada capa requiere volver upstream

