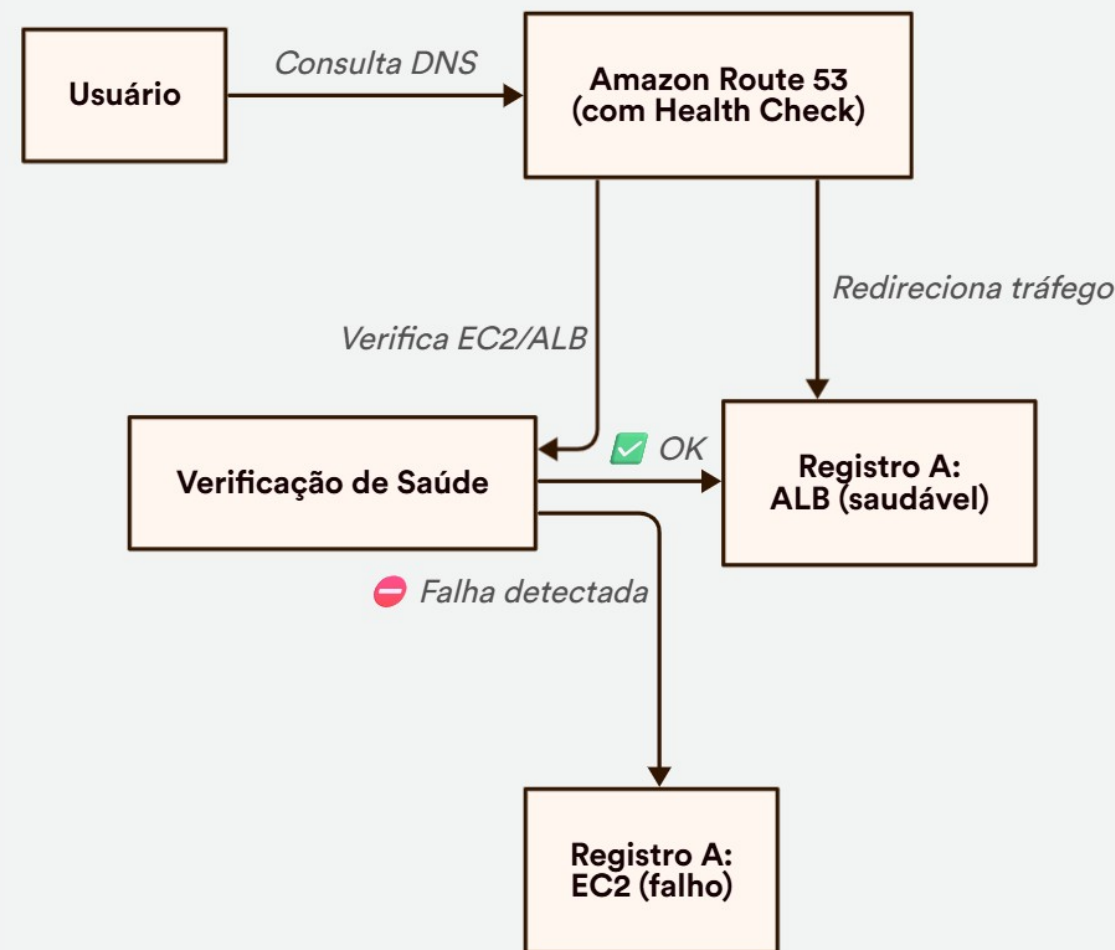


Route 53: HealthChecks e Políticas de Roteamento



Health Checks no Route 53

- Permitem verificar a integridade de endpoints (EC2, ALB, etc.) via HTTP, TCP ou HTTPS
- Se um recurso falhar, o Route 53 pode redirecionar o tráfego para outro registro saudável
- Podem ser usados em registros simples ou em conjunto com políticas de roteamento



Health Checks no Route 53

FUNÇÃO	DESCRIÇÃO	RELEVÂNCIA PARA A PROVA
Monitoramento de Endpoint	Verifica a integridade de endpoints como EC2, ELB, ALB, etc., via HTTP, HTTPS ou TCP.	✅ Questões sobre alta disponibilidade e failover
Failover DNS	Se um destino estiver indisponível, redireciona para outro registro saudável.	🔥 Muito cobrado em roteamento com failover
Health Check Simples	Verifica status via resposta de rede padrão (HTTP 200 ou resposta TCP).	✅ Questões sobre comportamento básico
String de Pesquisa	Permite verificar se uma string específica está presente na resposta (ex: conteúdo HTML).	🔍 Pode aparecer em questões avançadas
Health Check dependente	Permite que um health check dependa do status de outros (composição lógica).	⚠️ Mais raro, mas pode aparecer como distrator
Integração com CloudWatch	Alarme é acionado se a verificação falhar — útil para ações automatizadas.	💡 Pode ser abordado com automações e alertas
Regras de avaliação	Você define quantas falhas consecutivas marcam o endpoint como "ruim" (ex: 3 de 5).	📄 Aparece em perguntas com configuração de resiliência
Intervalo de Verificação	Definido como 10s ou 30s entre cada verificação.	⚙️ Pode aparecer como detalhe técnico para diferenciar alternativas

Política de Roteamento: Simple / Simples

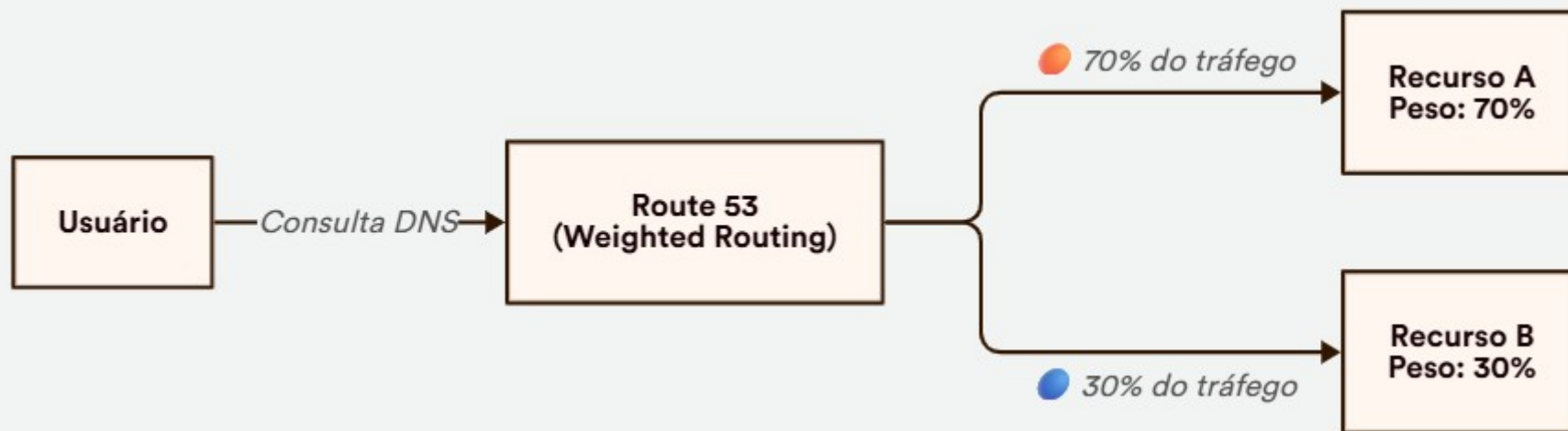
- Retorna um único valor configurado no registro — não tem lógica de decisão
- Ideal quando há apenas um recurso ou sem necessidade de controle de tráfego
- Pode ser usada com health check para failover simples



*** No Simple Routing, não há lógica de decisão – sempre é retornado o mesmo destino, ideal para cenários básicos ou sem variação de carga.*

Política de Roteamento: Weighted / Ponderado

- Distribui o tráfego com base em pesos atribuídos aos registros
- Ideal para migração gradual, testes A/B ou balanceamento controlado
- Pode incluir health check em cada peso para alta disponibilidade

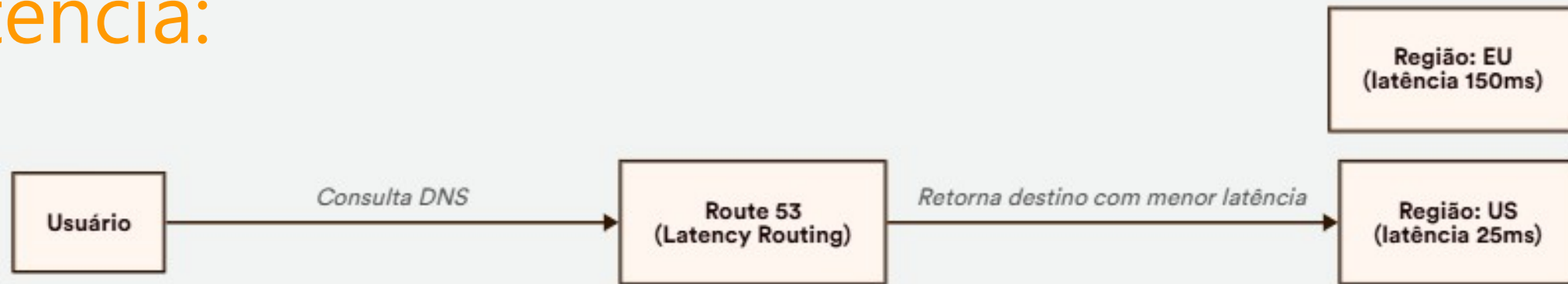


Políticas: Latency, Failover, Geolocation e Multivalor

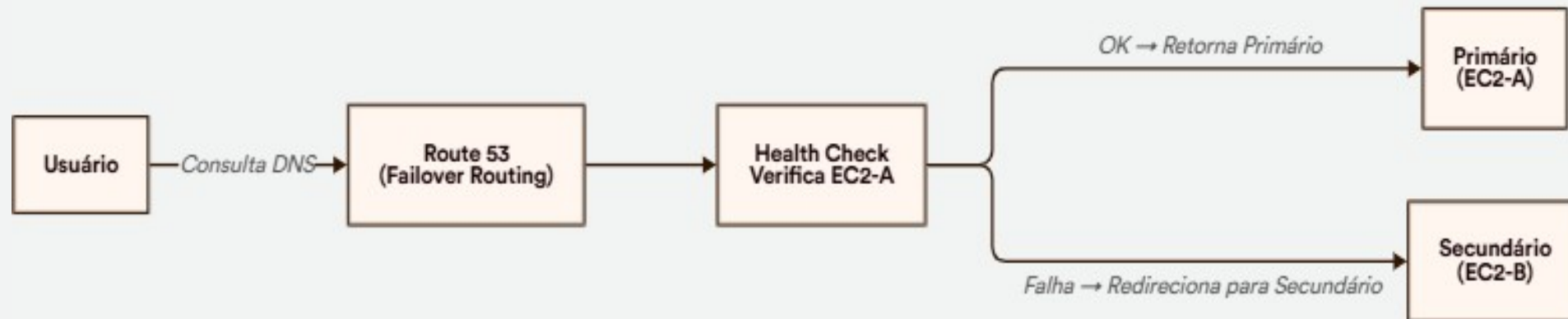
- Latency: envia o usuário para o recurso com menor latência percebida
- Failover: roteia o tráfego para o primário, e em falha, para o secundário
- Geolocation: direciona usuários com base na sua localização geográfica
- Multivalor: responde a consultas DNS com vários registros íntegros selecionados aleatoriamente

POLÍTICA	LÓGICA DE DECISÃO	QUANDO USAR	HEALTH CHECK?	DICA PARA A PROVA
Latency	Baseado na menor latência entre o usuário e o recurso	Melhor experiência ao usuário global	✗ (mas pode ser usado junto)	🌐 Palavras como "menor latência percebida" indicam essa política
Failover	Primário e secundário, com fallback se falha detectada	Alta disponibilidade entre 2 endpoints	✅ Obrigatório	🔴 "Em falha, redirecionar para..." = Failover
Geolocation	Com base na localização do usuário (país ou continente)	Conteúdo regionalizado ou requisitos legais	✗	📍 "Direcionar por país ou região" = Geolocation
Multivalue	Retorna múltiplos registros válidos aleatoriamente	Balanceamento simples sem ELB	✅ (opcional)	🎲 "Vários destinos aleatórios, mas íntegros" = Multivalue

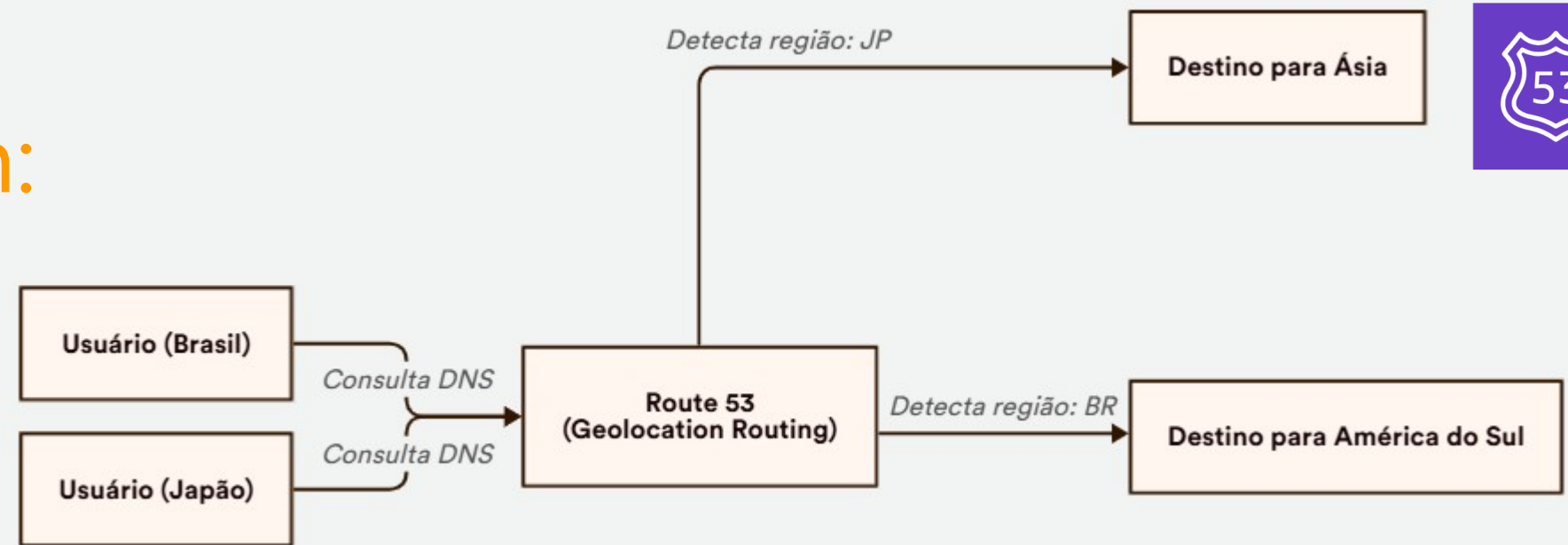
Latência:



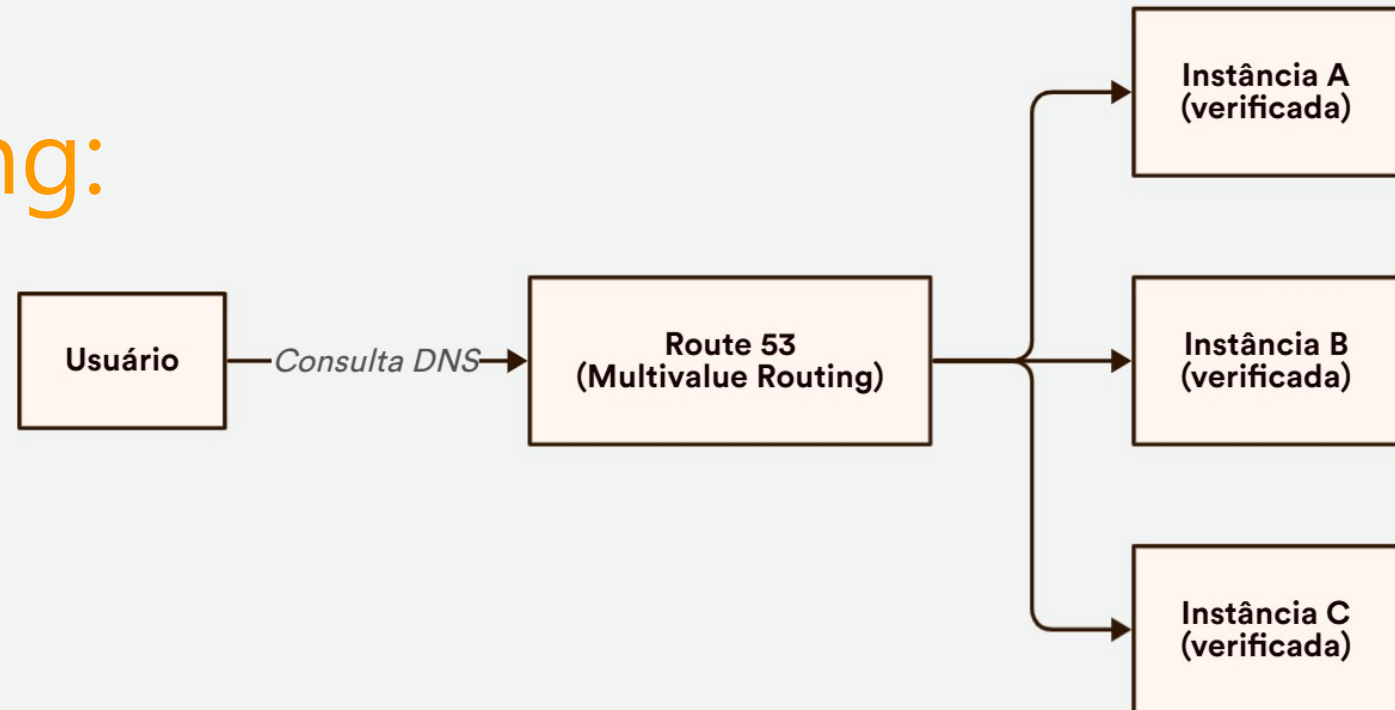
Failover:



Geolocation:



Multivalue Routing:



Boas Práticas e Dicas de Prova

- Health checks são essenciais para failover automático
- Simple = sem lógica; Weighted = balanceamento controlado
- Latency e Geolocation otimizam performance e compliance regional
- Failover requer no mínimo dois registros e health check ativo no primário
- A prova cobra bastante o uso certo da política para o cenário descrito