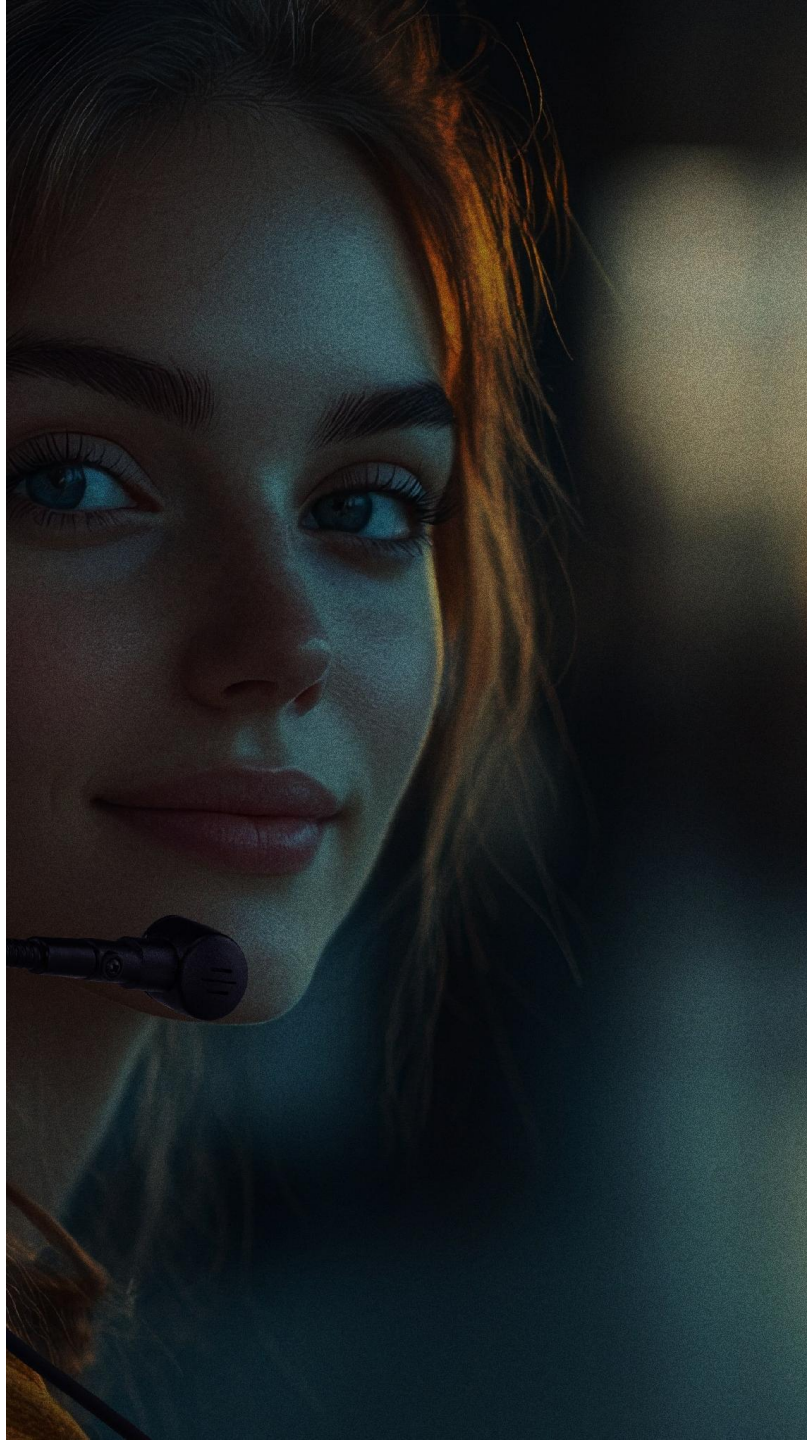
merapar

RTS RVOC

Running an intercom in the Cloud

A deep dive into security and network optimization of the RTS RVOC cloud intercom on AWS

Dutch Guild session · RVOC Engine · Elevate · Edge



INTRODUCTION

Who is talking?

Tom de Brouwer

Principal Architect · Merapar

Software delivery for SaaS & IaaS media platforms — from broadcast IP standards to the AWS infrastructure they run on. Based in Breda.

Off the clock: maintaining a 1977 Mini 1000, riding a Varadero

Broadcast IP media

ST 2110, NMOS, AES67 — and real-time comms: WebRTC, SIP, RTS intercom systems.

Cloud & AWS

12 AWS certifications; AWS Partner Network.

Open standards

Contributor to AES70/OCA; active in the IAMT Dynamic Software Licensing working group.

Background

Audio networking (Dante, AVB) at Bosch Security Systems — the RTS mothership — before IP media & cloud at Merapar.

Why this talk: consultant in the RVOC development team — the AWS architecture and security posture we walk through today is the one we build.

AGENDA

What we will cover

01 **RVOC platform**
Elevate, Engine, Edge — roles and trust boundaries

02 **Architecture & traffic paths**
Every plane: management, audio, hardware, media

03 **Deployment decisions**
Capacity, exposure, reach — one template

04 **Global Accelerator — the why**
Anycast edge entry vs. the public internet

05 **Global Accelerator — in RVOC**
What rides it: hardware ports AND media

06 **Security model**
Shared responsibility, least privilege, IMDSv2

07 **Identity, data & secrets**
Cognito, S3, Secrets Manager, credential lifecycle

08 **Network hardening**
Ingress CIDR, VPN/DX-only, private TURN, port ranges

09 **WebRTC media — relayed by design**
Relay-only posture, ICE, TLS/SRTP

10 **Security operations**
Signed updates, bulletins, recovery

merapar

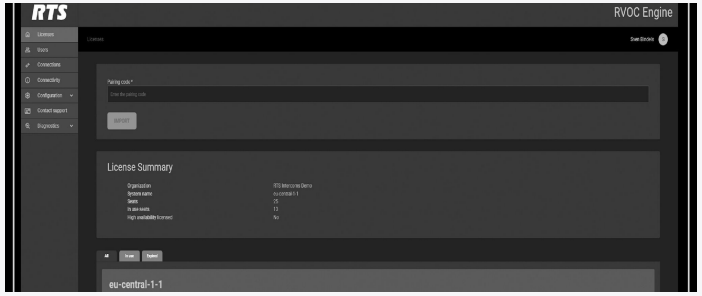
PLATFORM

Three components, two trust domains

RVOC Elevate

RTS-operated SaaS

Licensing portal at cloud.rtsintercoms.com: installer packages + signatures, license pairing codes, TCO calculator, security bulletins. The only RTS-operated piece.



RVOC Engine

Runs in YOUR AWS account

The intercom matrix + audio mixer, deployed from a supplied CloudFormation template. Phones home only to the RTS Backoffice for licensing, via NAT.

RTS Edge

iOS · Android · browser

Virtual keypanel on smart devices AND in the browser. Signaling over WSS via API Gateway; audio over WebRTC via ICE + STUN/TURN. Interops with RVON keypanels and ODIN/ADAM matrices.

Trust boundary: all intercom state, identity and media stay inside the customer account — Elevate never touches runtime traffic.

MILANO CORTINA 2026

NBC Sports Sets Its Own Record With Olympics Comms

By: Dan Daley, Audio Editor • Posted on: February 18, 2026 • 7 minute read

At the Broadcast Center in Stamford, three discrete intercom systems combine into a centralized infrastructure

Into the second week of Milan Cortina 2026, eight Olympic records and one world record have been broken so far. But the athletes aren't the only ones testing limits.

"This is the record-breaking stuff," says **John Pastore, senior director, broadcast communications, NBC Sports**, referring to the intercom infrastructure put together for NBC's production. It comprises 42 globally trunked intercom systems, using RTS's RVOC hybrid cloud-native intercom platform and RVON Voice Over Network, which allows full integration of an intercom system into an existing data network.

Essentially, three discrete intercom systems are combined into a single centralized intercom infrastructure for the entire NBC Sports

PLATFORM

RVOC Deployment options

RVOC Standalone

Standalone cloud intercom

Standalone intercom matrix of up to 2400 intercom ports. All communication non-blocking.

RVOC Hybrid

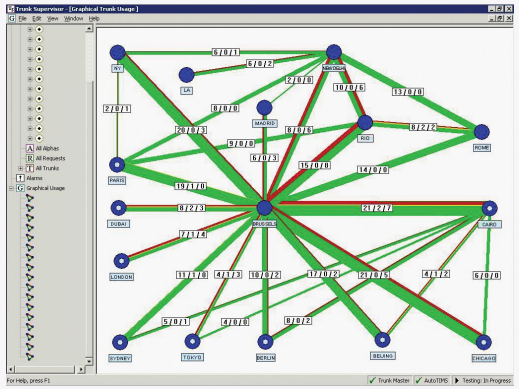
Hybrid Intercom including RTS ODIN frames

Stack a total of 9 frames (8 Hardware + 1 Software) to create an intercom up to 2400 ports.

+ Trunking

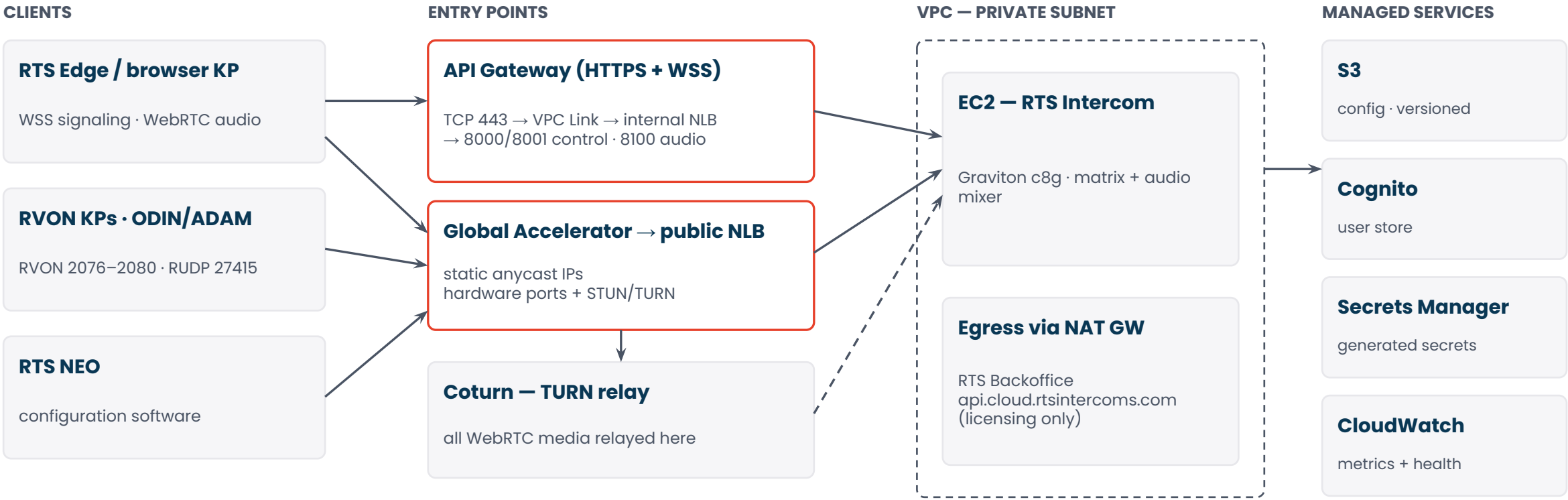
Trunk multiple intercoms together

Trunk multiple intercoms together and use trunk lines to communicate between different intercom systems



ARCHITECTURE

Essential deployment — traffic paths



Instances are cattle: all state lives in S3 (versioned) and Cognito; an EC2 replacement re-installs the signed .deb from the asset bucket on boot and reloads config. Route media Coturn → intercom inside the VPC to avoid data-egress charges.

DEPLOYMENT

One template, four decisions

Capacity

One number drives sizing: the maximum parallel audio connections (RVON + RTS Edge + tie-lines + SIP/NDI). The template picks the matching Graviton instance — size for peak, because resizing interrupts service.

Exposure

Decide up front how reachable the intercom is: fully public entry points, everything clamped to a single allow-listed address range, or no public termination at all — reachable only over VPN / Direct Connect.

Reach & media

One switch enables Global Accelerator for static anycast entry, and the stack deploys and wires the TURN relay for you — in a public or a private network segment, per your IT policy.

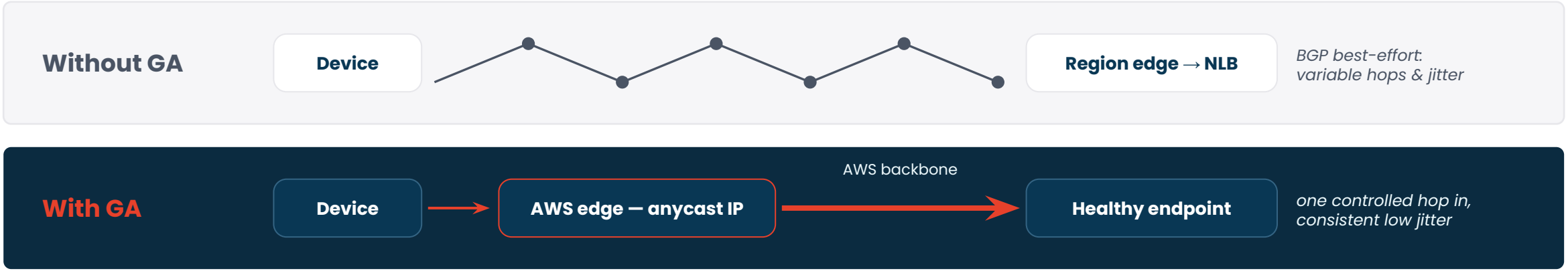
Integrations

Custom DNS and TLS via Route 53 + ACM, and an email provider (SES or SMTP) for invitation mails, password self-service and license notifications — recommended for any production deployment.

DEEP DIVE

AWS Global Accelerator — the traffic win

Same client, same intercom — a different path. GA hands traffic to the AWS backbone at the nearest edge instead of letting it ride the public internet all the way to the region.



Measure the gain from your own location: AWS Global Accelerator Speed Comparison — speedtest.globalaccelerator.aws

Latency & jitter

Traffic enters the AWS network close to the client — fewer internet hops for keypanel, trunk AND mobile media. Jitter budget: <10 ms (RTS Edge), 10/20/30 ms (RVON).

Stability for hardware

RVON panels and ODIN trunks are configured against fixed IPs — GA's static anycast IPs survive failover, so devices never re-point.

Health-based steering

GA + NLB health checks steer traffic to the healthy, active endpoint during failover — no DNS TTL wait.

DEEP DIVE

Why Global Accelerator wins — the mechanics

Onto the AWS network as early as possible

The same two static IPs are announced via anycast from AWS edge locations worldwide. Internet BGP therefore delivers the client to the NEAREST edge — only the first mile rides the public internet. Every transit provider and internet exchange after that point is bypassed.

…because the AWS global network is better

From the edge, traffic rides AWS's private backbone between edge locations and regions: capacity-managed, monitored, redundant. The public internet routes by BGP policy (AS-path, peering economics) — not by performance — and shares congested exchanges with everyone.

What that buys (AWS, p90, TCP)

-49% first-byte latency

-58% jitter — the number audio cares about

+60% throughput

Source: AWS Global Accelerator FAQ — measured by third-party real-user monitoring.

What it can't fix: the first mile is still the client's own ISP, Wi-Fi or 4G — GA takes over from the nearest edge onward.

DEEP DIVE

Global Accelerator in the RVOC stack — scope & rules

Rides GA — static anycast entry

- **RVON keypanels**
  TCP + UDP
- **ODIN / ADAM trunking (RUDP)**
  UDP
- **RTS NEO configuration**
  TCP
- **RTS Edge / browser-KP media**
  STUN/TURN

One template flag

One template switch — mandatory for High Availability deployments, optional for Essential; enable it whenever hardware integrations or internet media are in play.

Static IPs devices can trust

RVON panels and trunk configs point at fixed addresses. GA's static anycast IPs never change — no device re-pointing, no DNS TTL dependency, even when the active zone moves.

Region strategy

Still pick the region closest to the user majority; GA fixes the last-mile path, and RVOC adds built-in latency & jitter compensation on top.

Security posture

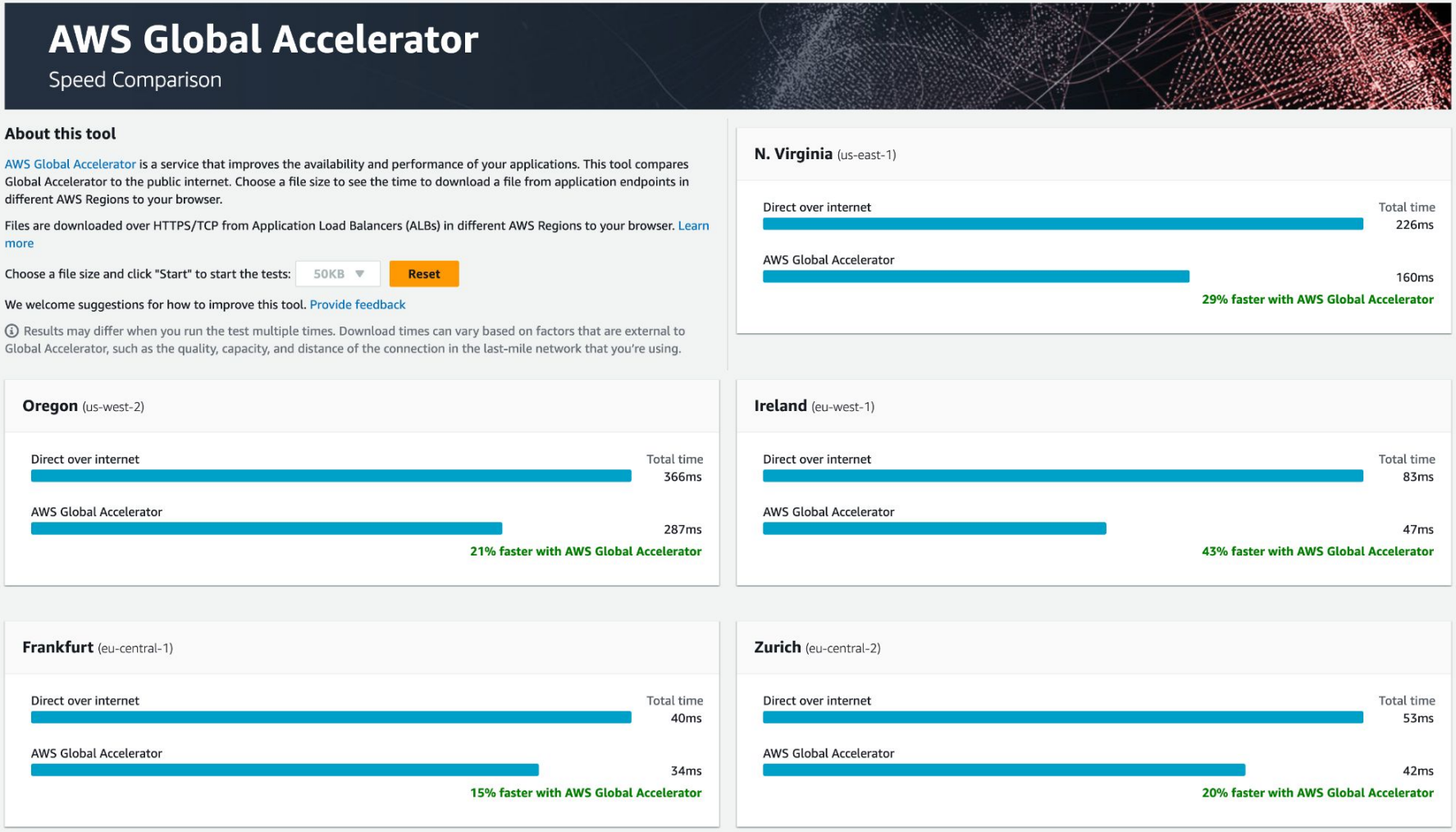
GA is one of the deliberately internet-exposed entry points — the IngressCIDRBlock clamp and NLB security groups behind it still apply unchanged.

Rule of thumb: everything latency-sensitive — hardware ports AND STUN/TURN media — enters through GA's static anycast IPs.

merapar

DEEP DIVE

Global Accelerator — measured, not claimed



How to read it

Compares direct internet vs. GA download time per AWS region

The gap grows with distance to the region and with lossy last miles

Same effect applies to keypanel, trunk and TURN media entering RVOC

merapar

SECURITY

Security model — shared responsibility, least privilege

AWS — security OF the cloud

Hardware, software, networking and facilities running the managed services.

You — security IN the cloud

OS, network & firewall configuration, client/server-side encryption, and the data itself.

Internet-exposed by design: Cognito · API Gateway · GA / NLBs · TURN. Everything else is private — the internal control & audio planes accept traffic only from the load balancers.

No root, ever

Deploys and runs without AWS root privileges — don't use root for any RVOC operation.

Least privilege by template

Every resource carries an IAM role or resource policy scoped to exactly what it needs: the instance role reaches its own S3 bucket, Cognito pool and secrets — nothing else. Zero manual IAM work, zero secrets to define.

Two application profiles

Administrator: RVOC Engine login + user management. Intercom User: RTS Edge only — no management-plane access.

Instance hardening

IMDSv2 fully supported, IMDSv1 not required. Hardened AMIs can be substituted (you own the monitoring tweaks).

SECURITY

Identity, data & secrets — nothing leaves the account

Amazon Cognito

identity

Stores email, first/last name, password per user. Reached over HTTPS with IAM roles; at-rest encryption per Cognito's own strategy. Bootstrap rule: the first login becomes the administrator.

Amazon S3

config & state

Intercom configuration + application-connectivity credentials, server-side encrypted, object-versioned for rollback. The instance is disposable precisely because this bucket is the source of truth.

AWS Secrets Manager

generated secrets

Deployment generates its own secrets at install (e.g. TURN long-term credentials) — you define none, and there's nothing to leak from a parameter file or pipeline.

Credential lifecycle

no rotation toil

Runtime access to other systems uses short-lived credentials only (instance IAM role); no manual key rotation exists as an operational task. SES integration likewise rides the instance role — zero stored SMTP-style creds.

SECURITY

Network hardening — pick your exposure posture

Hardware-integration (fixed set, every deployment)

Proto	Purpose
UDP	RUDP inter-intercom trunking
TCP	RTS NEO configuration
TCP	RVON keypanel
UDP	RVON
TCP/UDP	STUN / TURN

WebRTC media: UDP source ports come from rtc.ice port_range_begin/end (default 1024–65535, inclusive). Narrow it so SGs cover a known range (keep it ≥ max simultaneous streams). With relay-only media, clients face only the TURN server's ports — this range applies on the intercom side.

Exposure ladder — least to most restrictive

1 · Open + authenticated

Default: hardware ports public, management via Cognito-authenticated API Gateway.

2 · CIDR clamp

IngressCIDRBlock restricts all hardware protocols to one allow-listed range at the SG level.

3 · Private TURN

DeployCoturnNetworkLayer=Private puts Coturn behind an NLB in the private subnet — one extra hop, more control, noticeably higher TCO.

4 · No public termination

Disable the public NLBs entirely: hardware reaches the intercom only over site-to-site VPN and/or AWS Direct Connect. Add native IPv6 STUN/TURN if your estate is v6-only.

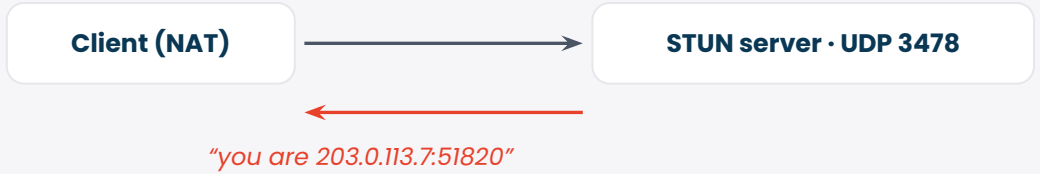
merapar

DEEP DIVE

STUN & TURN — how NAT traversal actually works

The problem: a client behind NAT doesn't know its own public address, and its NAT drops unsolicited inbound packets. Two parties behind NATs can't simply connect — someone has to help.

STUN — ask the mirror



- Reflection only: the server reports the public IP:port it saw — the server-reflexive address
- Used during setup; no media ever flows through it
- Only works if the NAT reuses that mapping for other destinations

TURN — rent a relay

allocation: a relayed address ON the server, kept alive by the client

- All media flows through the relay — both directions
- Always works: from the NAT's view it's ordinary outbound traffic
- Authenticated (long-term credentials) · costs 2× per-call bitrate at the relay

Why STUN alone isn't enough: symmetric NAT — the norm on enterprise firewalls and mobile carriers — creates a NEW mapping per destination, so the address STUN reports is useless to anyone else. TURN is the guaranteed path, and it's why RVOC relays all media.

merapar

DEEP DIVE

STUN & TURN — a short history

2003 — **STUN v1 — RFC 3489**

"Simple Traversal of UDP through NATs." Born from SIP/VoIP NAT pain. Promised to classify NAT types (cone, symmetric…) and predict mappings — real-world NATs refused to behave.

2008 — **STUN reborn — RFC 5389**

Renamed "Session Traversal Utilities for NAT": demoted from complete solution to a tool inside a larger algorithm. Same wire format, humbler job description.

2010 — **TURN & ICE — RFC 5766 / RFC 5245**

TURN standardizes the relay; ICE becomes the algorithm that gathers host, reflexive and relayed candidates, probes every pair, and picks a working path.

2011 — **WebRTC**

Google open-sources WebRTC; IETF/W3C standardization puts ICE, STUN and TURN into every browser — NAT traversal goes from VoIP niche to web-platform plumbing.

2018–20 — **The modern trio — RFC 8445 / 8489 / 8656**

ICE, STUN and TURN revised into the versions in force today (incl. IPv6 relaying). WebRTC itself becomes a W3C Recommendation in 2021.

today — **Coturn**

The open-source TURN/STUN server that grew out of rfc5766-turn-server — and the one the RVOC stack deploys and wires up for you.

Two decades of lesson-learning, condensed: don't try to outsmart the NAT — reflect when you can, relay when you must, and let ICE decide.

WebRTC media — relayed by design

Gather

Client + intercom collect ICE candidates: host, server-reflexive (STUN), relay (TURN)

Pair & probe

Standard ICE connectivity checks across the candidate pairs

Policy: relay only

RVOC runs with 'Use TURN relay only' — every media stream is forced through the relay

Flow

client ↔ TURN ↔ intercom. No direct client-to-intercom media path exists

Management & signaling

Management UI + Edge signaling terminate on API Gateway over HTTPS / WSS (443). Cognito & S3 access: HTTPS with IAM roles. Nothing management-plane travels in clear.

Why relay-only wins

The intercom's media plane is never directly reachable — clients only ever face the TURN server's ports. One controlled choke point with deployment-generated credentials, and identical behavior on every NAT type.

merapar

merapar

TAKEAWAYS

Communicate when it matters most

Secure path = default path

Least-privilege IAM, generated secrets, IMDSv2, versioned encrypted state — all template-enforced, all inside the customer's account.

GA earns its place

Static anycast IPs hardware can trust, backbone transport for latency, health-checked steering to the healthy endpoint — one template flag.

Exposure is a dial

CIDR-clamped ports, VPN / Direct Connect-only termination, private TURN — and all WebRTC media relayed, so the audio plane is never directly exposed.

A real trial matrix in ~10 minutes

One CloudFormation stack in your own account, RTS Edge on your phone, no license needed to install (only to test). Questions welcome.