One-way audio is one of those VoIP problems that feels simple until you are in the middle of it. The call connects, the phone shows it is answered, maybe you hear remote ringing and your partner hears you talk, and then at some point the audio one-sidedly collapses. Suddenly you are diagnosing audio paths across signaling, NAT, codecs, RTP ports, and https://www.avast.com/de-de/c-what-is-voip sometimes even device policies.

In practice, one-way audio almost always means the same thing: you have bi-directional call setup, but only one direction of the media path (RTP) is making it through. The trick is to identify which direction is blocked and then trace why.

Below is how I approach it in real networks, with examples, decision points, and the common traps that waste hours.

What “one-way audio” usually really means

VoIP calls typically use SIP for signaling and RTP for media. SIP confirms session parameters, like which IP and ports will carry audio. RTP carries the actual sound.

When audio is one-way, one of these is happening:

- Your SIP negotiation succeeded, so both endpoints believe they are sending RTP to the same place.
- But only one direction of RTP packets is reaching the other side.
- Often the endpoints are reachable for SIP, but RTP is blocked, translated incorrectly, or pointed at the wrong address.

The most revealing clue is who can hear whom. If the remote party cannot hear you but you can hear them, you are likely sending RTP out successfully, while your inbound RTP for your voice is blocked or misrouted. If you can’t hear them but they can hear you, the opposite direction is failing. Sometimes both sound directions fail, but that is more like “no audio,” not “one-way.”

That “direction” framing matters because many fixes are asymmetric: firewall rule changes, NAT traversal settings, endpoint media settings, and codec negotiation issues can each break only one stream.

Start with the fastest reality checks

Before diving into packet captures, I try to gather three pieces of information from the call itself and from the endpoints involved. This is where most one-way issues can be narrowed quickly.

First, confirm the scope. Is it one phone to one destination, one site, one carrier trunk, or all calls through a particular gateway? If you only see it on calls involving one external trunk or one provider, the fault is often on the carrier side, a border controller configuration, or routing through a specific NAT boundary.

Second, confirm whether the problem is consistent with a particular direction. If users at site A always cannot be heard by the far side, but they can hear inbound audio, you are likely losing outbound audio from that site or the far side cannot receive it.

Third, check whether the call is using the expected codecs and whether re-INVITEs or media renegotiation occur. Codec mismatch can cause “no audio” or “garbled audio,” but it can also masquerade as one-way if one side’s decoding fails more than the other side. It is less common than RTP path issues, but I keep it in mind.

A simple rule: if you can see RTP packets in one direction but not the other, you are in network or NAT territory. If RTP exists both ways, but audio is one-sided or distorted, shift attention to codec agreement and media handling.

Trace the media, not just the signaling

A SIP success does not guarantee RTP success. That is the whole story of one-way audio.

On the endpoint or PBX, look for a media/RTP status panel if it exists. Many systems show “RTP sent/received” counters per call, or they log the remote RTP address and port learned during the SIP exchange.

If your environment supports it, I prefer to validate with packet captures at a strategic point, usually the edge where NAT happens or where the VoIP media enters/exits the site. Capturing on the same interface as the SIP signaling can mislead you, because RTP might be on a different VLAN, a different interface, or even traversing a different firewall zone.

A practical way to decide where to capture

If you have a single NAT boundary per site (for example, one edge firewall that translates private RFC 1918 addresses), capture on the inside interface of the firewall and on the outside interface. If RTP flows are present on the inside but missing on the outside, the firewall or ALG behavior is stopping them. If RTP flows appear on the outside but not inside, the return traffic path is wrong.

If you do not have visibility, you can still do a “symmetry check” by comparing packet counters on both sides of the NAT boundary.

The NAT and RTP port range problem

The classic one-way audio cause is NAT. More precisely, it is NAT for RTP, and the most common failure mode is the firewall or NAT device not handling dynamic RTP ports correctly.

Why RTP is tricky

RTP typically uses a UDP port range that the endpoints negotiate in SIP or SDP. Many VoIP deployments use a range like 10000 to 20000 for RTP. If a NAT device or firewall allows SIP on UDP 5060 (or 5061 for TLS) but blocks UDP ports outside a narrow list, you get exactly what users report: calls connect, but media fails in one direction.

Even when the firewall allows RTP in general, incorrect NAT mapping or session helper behavior can cause only one stream to be translated properly. This can happen with SIP ALG feature sets, custom security policies, or when the endpoints are behind multiple layers of NAT.

Port rewriting and the “wrong destination IP” symptom

Another NAT-related symptom: one side sends RTP to the other’s private address, because the SDP contained the private IP or because a SIP proxy did not rewrite it correctly. The receiving endpoint hears nothing because it never receives packets sent to an address it cannot route.

In packet captures, this looks like RTP packets being sent to an RFC 1918 address from the public side, or from one NAT segment to another without translation.

The fix is usually to ensure that the PBX, SBC, or Voice over Internet Protocol gateway advertises the correct external media address, and that the NAT device performs consistent rewriting for the negotiated RTP ports.

The SIP ALG and “helpful” firewall features

Many administrators have turned on “SIP ALG” or “VoIP helper” features at some point because the firewall vendor says it improves SIP and NAT traversal. In some networks, it works. In others, it makes things worse, especially for RTP.

With ALG, the firewall can inspect SIP payloads and attempt to rewrite IPs and ports inside SIP or SDP. When it mis-parses the message, rewrites incorrectly, or conflicts with an SBC’s own behavior, you get media path mismatches.

Here is the pattern I see most often: you change nothing else, the carrier updates something or you upgrade firmware, and suddenly one-way audio appears. RTP counters drop one direction, but SIP still works.

The troubleshooting move is blunt but effective. If you control the firewall, test with SIP ALG disabled and ensure that your SBC or PBX is correctly advertising external media addresses. On some platforms you may need to disable the ALG on the specific policy or zone affecting VoIP traffic, not globally.

Because these features vary by vendor, I avoid a one-size-fits-all instruction. The key is to treat SIP ALG as “stateful and unpredictable” during troubleshooting, then validate RTP flow with packet captures.

Codec negotiation and asymmetric media handling

Codec mismatch is not the most common root cause for one-way audio, but it appears enough to warrant checks. Codec issues can lead to strange behavior:

- One side selects a codec the other side can negotiate but cannot decode properly for some reason (transcoding mismatch, missing licenses, codec disabled, or an endpoint firmware bug).
- A device chooses different codecs for different directions if it uses distinct media parameters for send and receive streams, some systems do.
- Media re-negotiation after early media can result in the endpoints switching codecs mid-call.

When codec problems cause “one-way,” you might notice that RTP packets still flow in both directions, but the receiving device reports decoding errors or silent frames.

What to look for

In logs, look for “codec selected,” “payload type,” “DTMF mode,” and any “media not compatible” warnings. If your VoIP system supports transcoding, verify that transcoding is actually occurring where you think it is.

If your provider or SIP trunk uses a preferred codec order, make sure your PBX or SBC aligns. For example, if you force a narrow set like G.711 only, you reduce variables. If you instead allow too many codecs, you increase the chance that one side will “agree” in SIP but mishandle the chosen payload.

Codec issues rarely cause RTP to disappear. So if your capture shows RTP missing entirely in one direction, codec is probably a secondary factor.

Firewall rules that allow SIP but not RTP

This is the simplest category, and it still happens in modern environments because security teams often treat “VoIP” like “just open SIP.”

SIP uses UDP and/or TCP 5060 or 5061. RTP uses a different UDP port range, often negotiated in SDP. If you only allow SIP, calls connect, and then audio fails. In some cases, stateful inspection means one direction works because packets trigger dynamic allowance in one direction but not the other.

If you see RTP packets leaving one endpoint but not arriving at the other, check the UDP port allowances between those specific IP addresses, not just from "the PBX network to anywhere."

Also consider that some deployments use multiple legs, such as: phone to PBX, PBX to SBC, SBC to carrier. One-way audio might be caused by the segment between SBC and carrier, even though phones-to-PBX calls look fine.

The "external media address" misconfiguration

Misconfigured media addresses are a frequent silent killer. It is easy to set the signaling address correctly, then forget that RTP needs a different advertised IP.

Common missteps include:

- Setting "externally reachable IP" for SIP but not the corresponding "external media address."
- Setting "local IP" to a private interface that is not reachable from the other side.
- Using a hostname that resolves differently for internal vs external DNS views.
- Relying on DNS that returns an IPv6 record when the far endpoint expects IPv4, or vice versa.

When media addresses are wrong, you can often confirm it by comparing the IP in the SDP offer or answer with the actual observed packet destination in captures. If the SDP says your public IP but the device actually sends RTP to a private IP, something in the rewriting path failed.

Sometimes the fix is to explicitly configure the correct public IP or the SBC address, and then disable any "auto detect" feature that guesses the external address.

Early media, ringback, and RTP timing quirks

Some one-way audio reports only occur after the call is answered, while others happen during ringback or early media. This can hint at how the endpoints treat media streams.

Early media is often transported differently, and some systems only open RTP on answer. If one device opens media early and the other waits, firewall state can create one-way behavior if the firewall allows packets in one direction after it sees a certain packet pattern.

If the one-way audio correlates with ringback or transfer events, validate that the firewall and SBC support the required media handling for early dialogs. Make sure that RTP isn't being blocked during the early dialog phase, then suddenly allowed only in one direction.

A focused troubleshooting workflow that usually works

At some point you have to stop random checking and run a structured path. Here is the workflow I use because it minimizes thrashing.

- Confirm which direction is broken by checking what each side can hear. Get a second test call with roles swapped if possible.
- Identify the RTP media endpoints (IP and port) from SIP/SDP logs on both sides, and compare with what your capture actually sees.

- Check NAT and firewall policy specifically for the RTP port range used by the system or SBC.
- Temporarily simplify the codec set to a known common codec and confirm RTP still flows in the failing direction.
- If you use an SBC or border gateway, verify it is the single point responsible for rewriting media addresses and that it is not fighting with firewall SIP ALG behavior.

You do not have to do all of those every time. The point is to quickly categorize the fault into “RTP not flowing” versus “RTP flows but audio is unusable,” then aim your effort accordingly.

Quick checklist for the first 15 minutes

If you want a short, repeatable start:

1. Note who can hear whom, and whether it changes after answer.
2. Check RTP received and sent counters for the failing call on the PBX or gateway.
3. Capture RTP at the NAT boundary and confirm packets exist in the missing direction.
4. Verify the SDP advertised IP and port match the expected external address.
5. Confirm firewall policy allows UDP for the negotiated RTP port range between the correct IPs.

That sequence often reveals the category within a few calls.

Case examples from the field

Example 1: “Calls connect, but the remote never hears us”

This happened to a site where SIP signaling worked fine through a firewall, but RTP was allowed only for a narrow port range because earlier deployments used a fixed range.

The fix was twofold. First, the PBX RTP port range configuration was updated to match what the firewall policy expected. Second, the firewall policy was broadened to cover the actual negotiated RTP range. After that, RTP started arriving in both directions consistently.

The reason it looked like “remote never hears us” is that inbound audio from the remote side triggered stateful allowances differently than outbound audio from the PBX, especially when the firewall did not see a matching outbound flow before the return traffic arrived.

Example 2: “Only one trunk, only one direction, after an upgrade”

After a firmware upgrade on an edge firewall, one-way audio appeared for calls involving a specific provider trunk. SIP logs showed dialogs established normally, but RTP packets were rewritten incorrectly inside SDP.

The immediate resolution was to disable SIP ALG (or its equivalent helper) on the policy tied to that trunk, and rely on the SBC for media address rewriting. RTP restored immediately.

The important lesson: do not assume “helper features” stay stable across firmware updates. If it breaks after a change, look at the stateful features first.

Example 3: “RTP flows, but users describe it as ‘muted’”

In this scenario, packet captures showed RTP packets moving both ways, so network traversal was not the blocker. Instead, logs revealed that one endpoint switched to a codec that the receiving side could negotiate but not

decode reliably for the media type used.

Restricting the codec set to a single common codec resolved it. After that, one-way audio disappeared, but it left behind a real process issue: the system's codec order had drifted because of an update or configuration template change.

Edge cases that mimic one-way audio

Some problems look like one-way audio but are actually different failure modes:

- **DTMF or call control audio**: Users sometimes think audio is missing when what they are actually missing is touch tones or system announcements.
- **Echo cancellation or media processing**: Some endpoints aggressively suppress audio if it thinks the far-end is silent. Mis-detected silence can make one direction feel muted.
- **Packet loss or jitter spikes**: Severe loss can make one direction sound "mostly gone." Captures will show RTP present but quality degrades. QoS and buffering settings can matter.
- **Wrong headset or local audio device**: Yes, it still happens. A headset microphone gain setting can make it seem like the far side cannot hear you, when the issue is only the send path on the user device.

I only mention these because they waste time. When I suspect local issues, I run a test using the same phone on a different line or a different port on the PBX to confirm the problem migrates with the call path rather than the endpoint.

Preventing recurrence

Once you fix one-way audio, the next pain is getting it back into a stable state with clear operational visibility.

I recommend standardizing on:

- A defined RTP port range for each system or ensuring the SBC controls it consistently.
- A documented firewall policy that includes the RTP port range, and that references the actual IPs used for media.
- Clear logs for RTP sent and received counters per call, and alerts if calls complete but media fails.
- A controlled change process for SIP ALG or firewall VoIP helpers, because those features can vary by firmware and policy scope.

Prevention is mostly boring work, but it saves weekends.

When you need help from the carrier or vendor

Sometimes the problem is outside your control, especially when your carrier provides a hosted SIP trunk or passes traffic through their own media gateways.

In those cases, you can still make progress if you gather the right evidence:

- Provide call identifiers, timestamps, and the remote trunk details.
- Share SIP dialog information, including the SDP offer and answer if your system exports it.
- Provide your RTP capture evidence showing which direction is missing at the boundary you control.

Carriers are usually faster when they can see the problem is “RTP packets never arrive” rather than “audio is one-way.” Clear, directional evidence helps them map it to their side: whether they allowed RTP ports, whether their NAT traversal is correct, or whether their SBC is rewriting SDP properly.

Final thoughts: treat it as a media path problem

The frustrating thing about one-way audio is that it tempts you to look only at SIP. The call setup is the easy part. The real work is RTP.

When you approach it as a question of which direction of RTP is failing, the troubleshooting becomes almost mechanical. First, confirm the direction. Next, verify whether RTP exists where it should. Then fix the place where IP and port expectations diverge, usually at NAT, firewall rules, or media address advertisement.

If you want, tell me what your setup looks like (PBX or SBC brand, whether endpoints are behind NAT, whether you use a carrier SBC, and a rough RTP port range). I can suggest the most likely failure points and where to capture first without burning time.