

Edge Device Network Configuration

Connecting AI to an isolated camera network — the connectivity test, the two supported designs, and how to verify them.

Visionify Inc. · Updated 2026-08-10 · visionify.ai/resources/edge-device-network-configuration

Almost every site that deploys computer vision runs into the same question in the first week: **the cameras are on one network, the internet is on another, and the edge AI device needs both.**

This guide is written for IT, OT and physical security teams preparing a site for an edge AI deployment. It explains why the conflict exists, the one test that determines which design applies, the two supported network configurations, and how to verify the result. It applies to any edge video analytics appliance — the examples use Visionify's edge device, but the network principles are the same for any RTSP-based analytics platform.

In a hurry? If you want a configuration that is guaranteed to work without testing anything first, use **Option B — dual-homing**, below. It is valid whether or not the camera VLAN has internet access.

Why camera networks are isolated in the first place

IP cameras are long-lived, rarely patched, and historically full of firmware vulnerabilities. Network segmentation is the standard mitigation, and it is baked into most security frameworks — IEC 62443 zones and conduits, NIST SP 800-53 boundary protection, and the network segmentation requirements in most cyber insurance questionnaires.

So a typical site ends up with:

- A **camera VLAN** (often `192.168.x.x`), carrying cameras, one or more NVRs, and a VMS workstation. Frequently no default gateway, no DNS, and no route off the segment.
- A **corporate / IT network** (often `10.x.x.x`), carrying user devices and servers, with managed internet access through a proxy or firewall.

That is good security design, and nothing in this guide asks you to undo it. The goal is to give one device — the edge appliance — a foot in each world, without exposing the cameras to the internet.

What the edge device actually needs

An edge AI device performs inference locally. Video never leaves the site; only event metadata and small evidence clips go to the cloud. That means it needs two conversations at once:

Direction	Destination	Protocol / Port	Purpose
Local (LAN)	Cameras / NVR	TCP 554 (RTSP)	Pull live video streams
Local (LAN)	Cameras / NVR	TCP 80, 443 (ONVIF)	Camera discovery, profiles, PTZ
Local (LAN)	Cameras / NVR	UDP 3702 (WS-Discovery)	Optional ONVIF auto-discovery
Outbound	Management + AI cloud	TCP 443 (HTTPS)	Remote management, event upload, updates
Outbound	DNS resolver	UDP/TCP 53	Name resolution
Outbound	NTP server	UDP 123	Clock sync (required for TLS and event timestamps)

No inbound ports from the internet are required. All cloud communication is outbound-initiated from the device, so the device never needs a public IP, a port forward, or an inbound firewall rule.

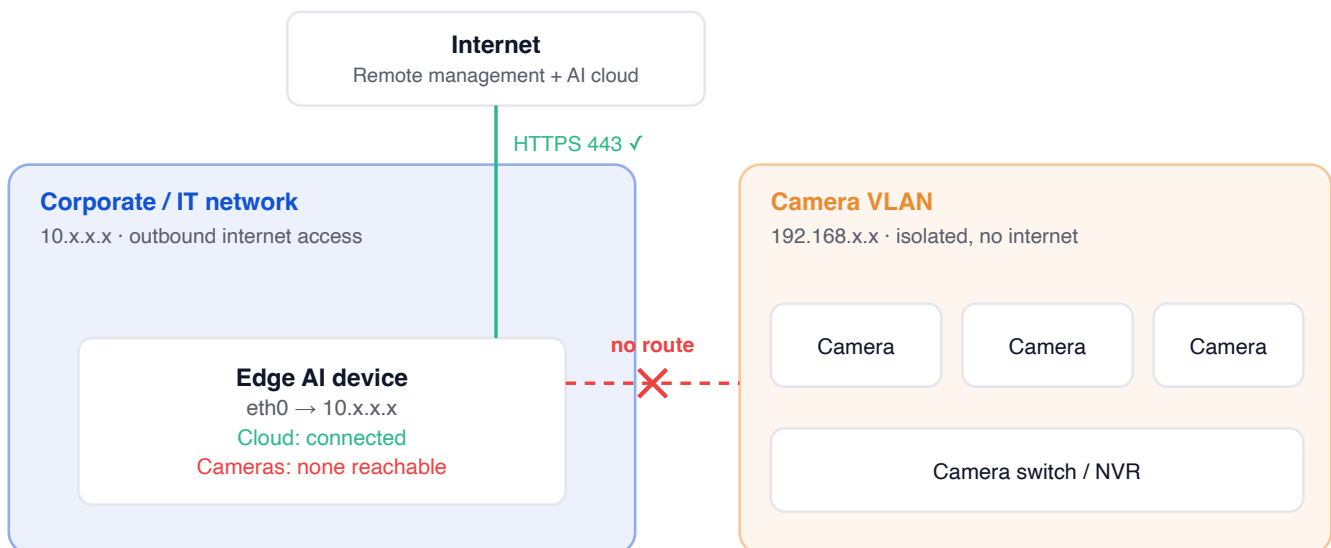


Figure 1 — The common starting point: the edge device reaches the cloud but has no route to the camera VLAN.

The starting point at most sites looks like the diagram above: the device is plugged into a corporate port, so it is online and remotely manageable — but it cannot see a single camera, and no video is being processed.

Step 1 — Test whether the camera VLAN has outbound internet access

Run this test before changing anything on the device. It answers a single question, and the answer determines which of the two configurations you should implement. The two are not interchangeable — choosing wrong can leave the device unreachable and require a site visit to recover.

Connect a laptop to a spare port on a **camera-network switch** — the same VLAN the cameras are on — and run all three checks:

```
ping 8.8.8.8 # raw internet reachability (ICMP)
nslookup www.microsoft.com # DNS resolution
curl -I https://management.azure.com # HTTPS on port 443
```

On Windows, use `Test-NetConnection 8.8.8.8 -Port 443` in PowerShell in place of `curl`.

Run all three. Many networks block ICMP but permit HTTPS, so **a failed ping alone is not conclusive — the HTTPS test is the one that matters.**

Result	What it means	Go to
All three succeed	The camera VLAN has usable outbound internet	Option A
HTTPS fails (or DNS fails)	The camera VLAN is isolated, as designed	Option B
Not sure / can't test	Use the design that works in every case	Option B

Do not implement Option A if the HTTPS test fails. The device will lose its cloud connection the moment the cable moves, and it cannot be recovered remotely.

Option A — Move the edge device onto the camera VLAN (camera VLAN *has* internet)

The device's Ethernet cable moves from the corporate switch to a camera-network switch port. It keeps a single connection and uses the camera VLAN for both jobs: pulling RTSP streams locally and reaching the cloud over HTTPS.

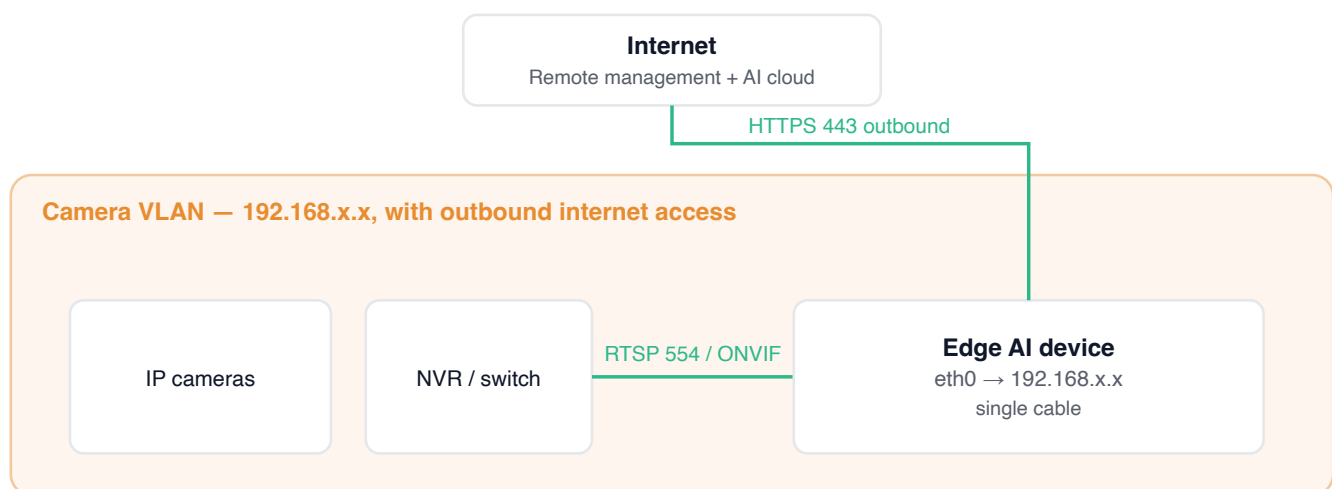


Figure 2 — Option A: the camera VLAN has outbound internet, so one cable covers both jobs.

What the network team does:

1. Move the Ethernet cable to a free port on a camera-network switch, on the same VLAN as the cameras.

2. Assign the device a **static IP or DHCP reservation** on the camera subnet, with the correct default gateway and DNS servers. Never leave an edge device on a dynamic lease — the IP appears in camera ACLs, monitoring and support tickets.
3. Nothing further. The Step 1 test has already confirmed the segment permits the outbound traffic the device needs.

Trade-offs: simplest cabling and a single interface to manage, but it puts an internet-connected device on the camera VLAN, which may conflict with your segmentation policy. It also depends on the camera VLAN's internet path staying available — if that changes later, the device goes offline.

After the cable moves, the device will drop off the cloud briefly and reconnect on its new address. If it has not returned within roughly ten minutes, the camera VLAN is likely blocking a required endpoint. Move the cable back to its original port and switch to Option B.

Option B — Dual-home the edge device (no internet on the camera VLAN)

The device gets two network connections at once: its built-in Ethernet port to the camera VLAN for video, and a second interface — a USB Ethernet adapter or Wi-Fi — to the corporate network for outbound HTTPS. This works even when the two networks are completely isolated from each other, and it requires **no routing or firewall changes on either side**.

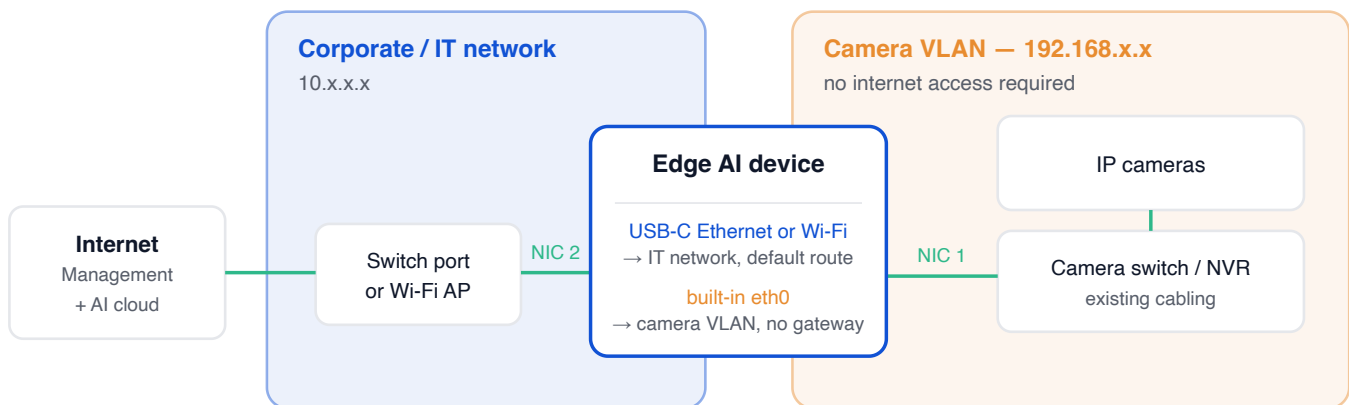


Figure 3 — Option B: two interfaces. One to the camera VLAN for video, one to the IT network for outbound HTTPS.

What the network team does:

1. **NIC 1 — cameras.** Ethernet from the device's built-in port to a free port on a camera-network switch, on the same VLAN as the cameras. Assign a static IP or DHCP reservation on the camera subnet, with a subnet mask **but no default gateway**.

2. **NIC 2 — internet.** Ethernet from a USB-C Ethernet adapter to a corporate network port — exactly how the device is connected today. Assign an IP, **default gateway and DNS servers** on this interface. Corporate Wi-Fi credentials work equally well if a second drop is inconvenient.

The one rule that matters: exactly one default route. The internet-facing interface carries the default gateway; the camera-facing interface carries only a directly-connected subnet route. Two default routes is the most common cause of an edge device that connects, then intermittently drops off the cloud. Your Visionify contact configures this on the device itself — the network team only needs to supply the addressing.

Trade-offs: one extra cable or adapter, in exchange for keeping the camera VLAN fully isolated. This is the recommended design for most sites, and the safe default when the Step 1 test cannot be run.

Choosing the second interface

The edge device has **USB-C ports only**, so it needs a USB-C to Gigabit Ethernet adapter — a USB-A adapter will not fit without a further adapter. Any standard model with a widely supported chipset (Realtek RTL8153 or ASIX AX88179) works with no driver installation. A commonly used example is the Anker PowerExpand USB-C to Gigabit Ethernet adapter, which is listed as Linux-compatible.

Please supply the adapter locally — it is a commodity part and sourcing it on site is faster than shipping one.

Check 802.1X and MAC allowlisting before install day. A USB Ethernet adapter presents a different MAC address from the device's built-in port. If your corporate network authenticates ports by MAC or 802.1X, that address must be authorised in advance. Ask your Visionify contact for the MAC address ahead of the install.

Option C — Route between the VLANs (only where inter-VLAN routing already exists)

If your network already routes between the corporate and camera segments, and policy allows a narrow exception, the device can stay exactly where it is and reach the cameras through the firewall. This avoids any cabling work, but it is the option that most often stalls — it needs security approval, and at sites where the camera VLAN has no Layer 3 route at all, there is nothing for a rule to permit.

The minimum ACL, scoped to the edge device's IP address only:

Source	Destination	Ports	Purpose
Edge device IP	Camera subnet	TCP 554	RTSP streams
Edge device IP	Camera subnet	TCP 80, 443	ONVIF control
Edge device IP	Camera subnet	ICMP echo	Reachability checks

Note that ONVIF WS-Discovery (UDP 3702) is multicast and does not cross a routed boundary — cameras must be added by IP address rather than auto-discovered in this design.

Step 2 — Verify the configuration

Once the chosen option is in place, these checks confirm both halves are working. Visionify runs them remotely and shares the results with you:

```
ip route                # exactly one default route, via the internet-facing NIC
ping -c 3 192.168.x.x   # camera reachable on the local VLAN
curl -I https://<cloud-endpoint> # outbound HTTPS working
ffprobe rtsp://192.168.x.x # stream opens and negotiates successfully
```

A healthy device shows: one default route, cameras answering on the camera subnet, HTTPS returning a response, and `ffprobe` reporting a video stream with resolution and codec.

Troubleshooting

Symptom	Likely cause	Fix
Cloud connects, no cameras visible	Device is on the corporate VLAN only	Implement Option A or B
Cameras visible, cloud offline	Camera-facing NIC has a default gateway	Remove the gateway from the camera NIC
Connectivity drops intermittently	Two default routes competing	Leave a gateway on the internet-facing NIC only
Second NIC never comes up	802.1X / MAC allowlisting on the corporate port	Authorise the adapter's MAC address
Camera pings but RTSP times out	Wrong RTSP port, or per-camera ACL	Confirm TCP 554 and the camera's stream path
RTSP prompts repeatedly for credentials	Special characters in the password	URL-encode the password, or use a simpler one
Device IP changes after a reboot	Dynamic DHCP lease	Convert to a static IP or DHCP reservation
TLS errors to the cloud	Clock drift	Confirm the device can reach an NTP server

What to prepare before install day

Have this ready and the configuration usually completes in under an hour:

- **Camera IP range or list** of camera IP addresses, and the VLAN ID.
- **RTSP / ONVIF credentials** — ideally a dedicated read-only account rather than the administrator login.

- **Addressing for each interface:** static IP or DHCP reservation, subnet mask, and — for the internet-facing interface only — gateway and DNS servers.
- **Confirmation of 802.1X or MAC allowlisting** on the corporate network, if in use.
- **A free switch port** on the camera VLAN, and one on the corporate network if dual-homing.

Summary

1. **Run the connectivity test** on the camera VLAN first — HTTPS is the check that matters.
2. **If the camera VLAN has internet access**, move the device onto it (Option A). One cable, simplest setup.
3. **If it does not** — or if you cannot test — dual-home the device (Option B). One NIC to the cameras, one to the corporate network. Works in every case.
4. **Keep exactly one default route**, on the internet-facing interface.
5. **Where inter-VLAN routing already exists**, a narrow ACL from the device's IP to the camera subnet (Option C) is also valid.
6. **Ask for help before you change anything.** Visionify can join a short call while the cable is moved and verify the result remotely within minutes.

© Visionify Inc. · The latest version of this guide is available at visionify.ai/resources/edge-device-network-configuration · Questions: support@visionify.ai