Verkada

Camera Setup Best Practices

Table of contents

General overview

Each Verkada camera is architected to automatically connect to the Verkada cloud via a secure bi-directional communication channel in order to:

1. Upload footage, hyperzooms, thumbnails, archives, and additional device data, in addition to generating alert notifications for specific device events (online/offline, tamper, occlusion).
2. Download firmware and update settings from Command (such as security enhancements and optical zoom settings).

By default, Verkada cameras do not connect to on-premise 3rd party NVRs or use insecure protocols (like HTTP or RTSP). If required, RTSP can be enabled for streaming footage into existing infrastructure or 3rd party analytics applications. Integrations with SIEMs can be set up via the Verkada cloud.

To learn more about our security and encryption solutions – including security at the device, cloud, product, and application levels – see our security page.

Installation and setup

Powering and connecting the cameras

Verkada cameras leverage Power over Ethernet (PoE) for power and communication over your LAN (local area network). In most cases, cameras will connect directly to an access switch that supports the necessary PoE standard. PoE requirements vary by device, and can be found here.

The switchport must be configured as an access port and all cameras will negotiate at full duplex.

Helpful tip

Most switches have a PoE budget that dictates the total amount of power that can be supplied across all switch ports. Always ensure that there is sufficient PoE budget for all devices plugged into the switch. If the PoE budget is exceeded, individual cameras may not power on or perform properly in demanding environments.

All Verkada cameras have infrared (IR) LEDs used to illuminate the field of view at night, leading to higher power draw. If a camera is powered on during the day but drops offline at night, the issue is likely related to insufficient PoE budget.

Some network switches do not provide PoE. In this scenario, PoE injectors can be inserted between the camera and the switch. The PoE injector will provide power to the camera, while the switch port will be used for LAN communication. PoE injectors can also be used when a camera requires more power than an individual switch port is able to provide.

When replacing an existing camera system that leverages coaxial cable, it might not be feasible to re-cable with Ethernet. Converters can be used to run Verkada cameras on PoC (Power over Coax) as detailed here.

Media converters can also be used to convert fiber runs to Power over Ethernet as detailed here.

For camera deployments in locations without wired internet, Verkada offers both cellular and Wi-Fi gateways. These devices require power but provide internet connectivity using LTE or neighboring Wi-Fi networks.

IP addressing and subnetting

When Verkada cameras are powered on, they will use DHCP to ask for a local IP address. If there is a requirement to have fixed IP addresses, it is recommended to configure DHCP reservations on the network DHCP server, a process of matching a reserved IP address with a camera's MAC address. Verkada can provide a list of MAC addresses associated with an order upon request.

Alternatively, cameras can be configured with a static IP address after initially staging the cameras on a DHCP network. Learn how to configure static IP addresses for Verkada cameras here.

Helpful tip

Using a dedicated Verkada VLAN adds an extra layer of security and will mitigate performance issues that arise when too many devices share the same broadcast domain. This will also allow you to adequately mark the traffic from a QoS perspective to make sure it is prioritized in favor of bulk traffic.

For organizations with strict network security requirements, Verkada cameras support port-based device authentication using the IEEE 802.1x protocol. Customers can upload their own certificates, private key, and create an (optional) private key password. Learn how to set up port-based device authentication for Verkada cameras here.

802.1x prevents non-controlled devices from connecting to the LAN – one key principle of Zero Trust Network Access (ZTNA). To learn more about Verkada's commitment to security and our "zero trust" approach, see our trust hub.

Firewall settings

Verkada cameras require access to various endpoints to ensure they can communicate with Command and all features are accessible. The general domains to allow (applicable for all organization regions) are:

- *:4100 – TCP/UDP on local network
- *.verkada.com – UDP/123 + TCP+UDP/443
- time.cloudflare.com – TCP/4460 + UDP/123
- *:amazonaws.com – TCP+UDP/443

Verkada cameras initiate communication to the cloud from within your network, so there is no need to set up port forwarding. In addition, because the cloud is acting as the VMS, there is no need to utilize a client VPN when viewing footage remotely.

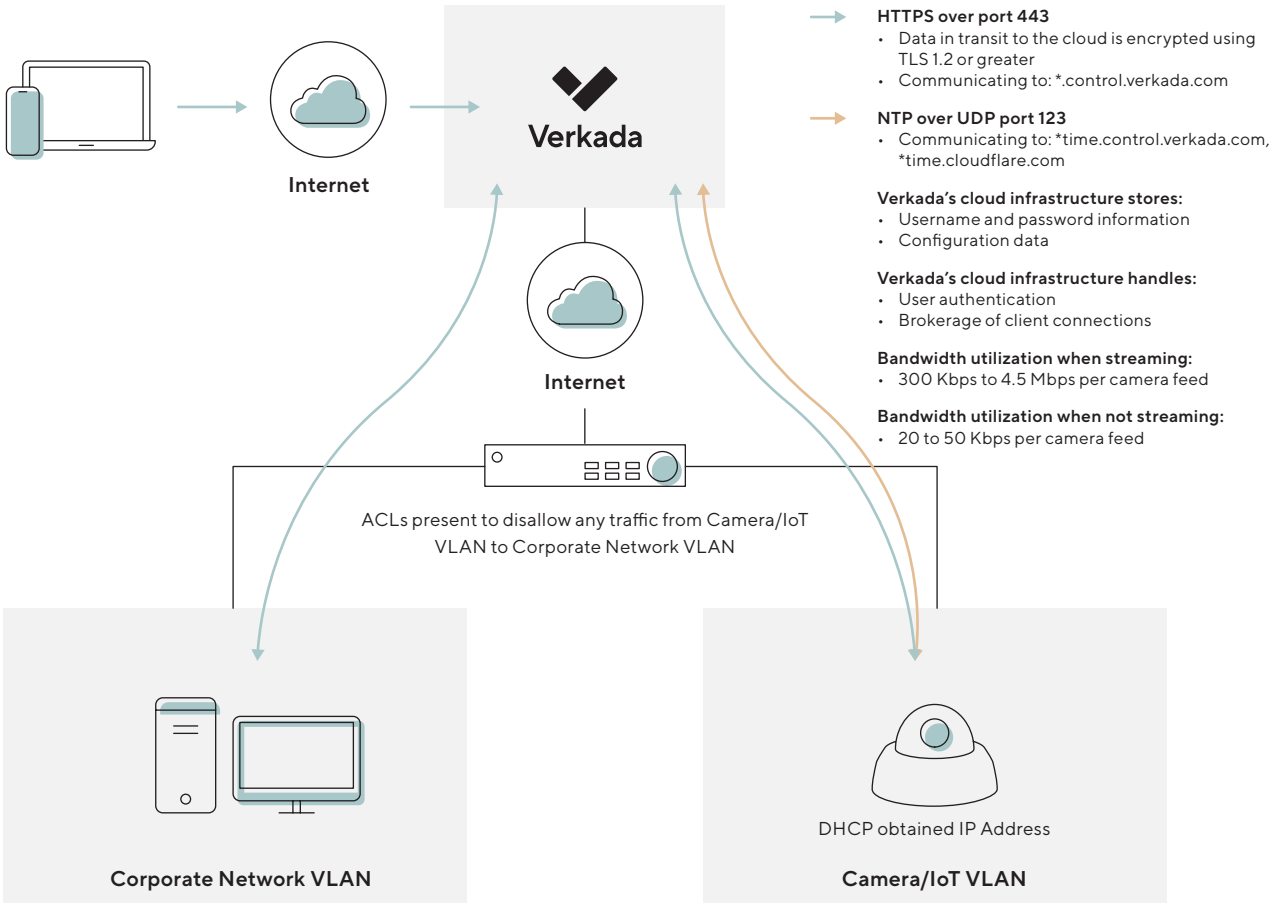
If required endpoints are blocked, cameras may not boot properly. This will be indicated by the LED status light indicator on the camera. View a full list of LED status light patterns and the associated LAN or WAN errors here.

Helpful tip

The easiest way to verify that your network settings meet the requirements for Verkada devices is to run Verkada's network testing tool, which can be found here.

As best practice, we recommend setting specific rules whitelisting the Verkada domains used, as opposed to allowing all traffic. A comprehensive list of required domains by region can be found here.

Below, we have outlined camera/IoT VLAN separation and traffic flows:



All Verkada cameras use AWS public key infrastructure (PKI) to ensure they only talk to the Verkada cloud, so SSL decryption needs to be turned off when inspecting Verkada traffic. Any attempt to enable it will break the communication.

For example, this article outlines how to deploy Verkada devices where ZScaler is used to inspect network traffic.

Bandwidth considerations

Verkada's system architecture is designed to unlock the benefits of the cloud in a bandwidth-efficient way. Cameras typically use a small amount of bandwidth at rest (20-50 kbps), but values can vary depending on factors like the amount of motion captured and types of analytics being leveraged.

Because of this, we recommend that you review the current utilization of your ISP links in order to avoid scenarios where cameras are deployed in an already oversubscribed environment. This can lead to a wide range of issues, such as remote streaming not working properly or the camera having issues downloading firmware updates.

Note that cameras need to be able to reach the Verkada cloud (as described in the firewall settings section above) and will work if you are using DIA (Direct Internet Access) or centralized breakout of a remote main site (when using MPLS to connect). If the site has both direct Internet links but also MPLS, we recommend setting up routing policies to prefer DIA and use MPLS as a backup (if Internet breakout is possible).

When trying to compute the bandwidth requirements for a camera, you need to account for:

1. The bandwidth consumed at rest (when nobody is viewing footage). This tends to be between 20-50 kbps, but can reach upwards of 200 kbps with advanced analytics turned on (especially in a scene with a lot of activity).
2. The bandwidth needed when footage is being viewed through the cloud. This varies by quality, resolution, and camera type, and can range from 300 kbps to 4.5 mbps. A full breakdown by camera model can be found here.

A few important things to consider:

- Verkada cameras utilize variable bitrates by default, allowing for video to be recorded at a lower bitrate during less complex scenes and a higher bitrate during more complex scenes. This ensures high-quality video when it matters most while optimizing storage and bandwidth usage.
- When multiple users access the same live feed remotely, AWS will multiplex the video in the cloud and provide multiple download streams, resulting in only one uplink stream being generated. As a result, bandwidth consumption is the same regardless of the number of users streaming the camera.
- When watching historical video, the bandwidth used will increase linearly with the playback speed. For example, playing back historical footage at 2x speed will lead to a 2x increase in bandwidth consumption.
- If set up for constant upload, cloud backup bandwidth consumption is identical to streaming bandwidth consumption.

Tips for conserving bandwidth, if required:

- Disable advanced analytics that are not being used.
- Use the cloud backup schedule to upload footage outside of working hours (or disable cloud backup entirely).
- Ensure users on the same LAN are able to locally stream footage.
- Set the default stream viewing quality to SQ (users can still change the viewing quality to HQ as needed).
- Low bandwidth mode can be enabled to decrease the resting bandwidth consumed by Verkada cameras by up to 75% and the streaming bandwidth by up to 33%, at the cost of a slight reduction in video quality and video scrubbing experience. Find more information here.

Local streaming

When accessing a camera's live stream, the device running Command prioritizes streaming over the LAN. If the private IP address of the camera is reachable and proper domains are allowlisted on the network, the client device will establish an HTTPS connection with the camera to directly obtain the live feed. This means that the camera does not need to upload the data to AWS just for it to return to the same local network. This ensures that ISP bandwidth is not overutilized and latency is minimal.

Requirements for local streaming:

- The accessing device must be able to reach the camera's private IP address.
- Port 4100 TCP/UDP must be open bidirectionally between the client and the camera.
- No proxies can be present between the client and the camera.
- Necessary domains are allowlisted on the network (view the complete list here).

Helpful tip

The accessing device does not have to be in the same VLAN as the device to access them, but routing between VLANS must be possible.

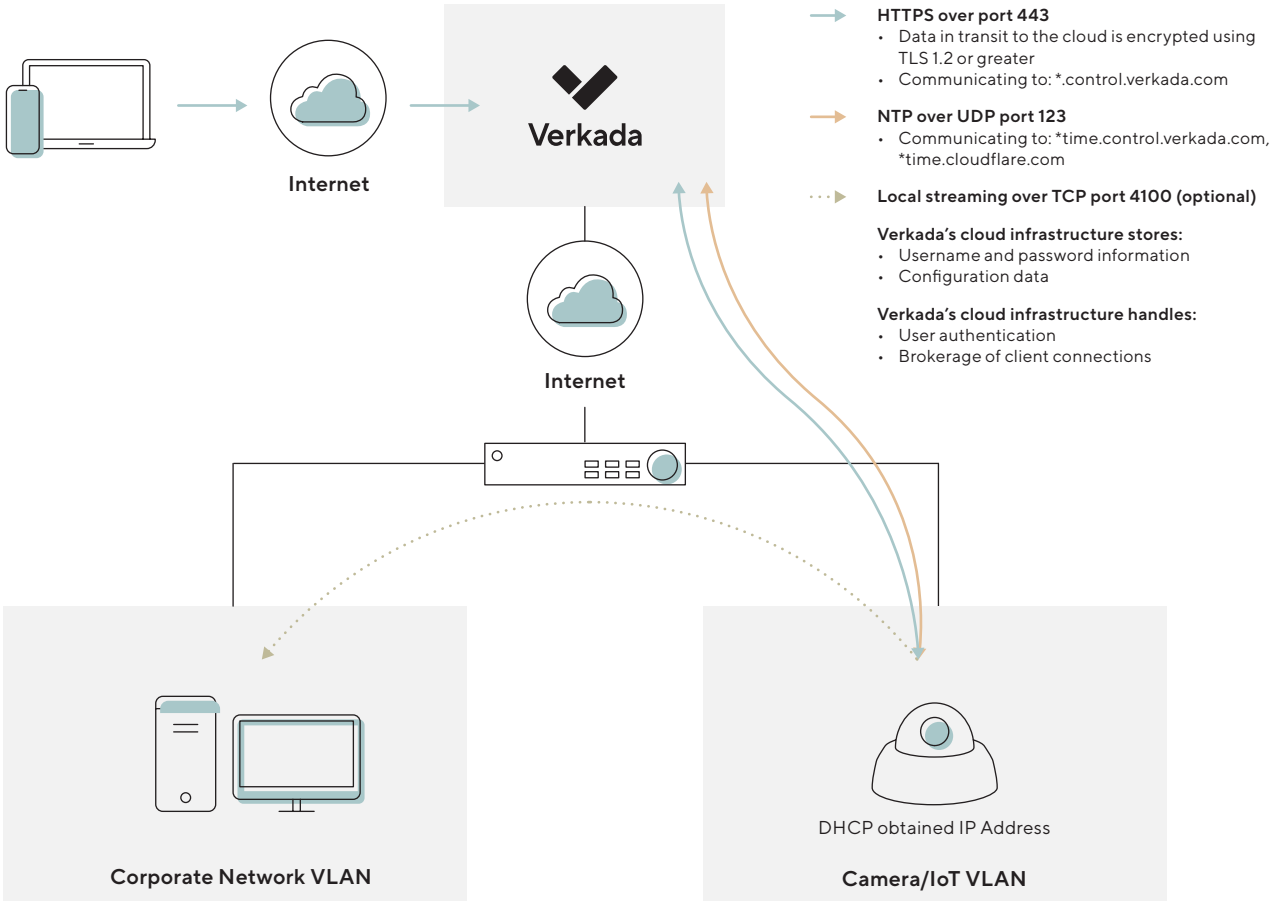
There are two visual indicators that indicate when live video is being streamed directly from a camera over the local network:

1. A white border around the green dot located next to the timestamp (in the top left-hand corner of the video feed)

2. The presence of the word "LOCAL" next to "HQ" or "SQ" (in the bottom left-hand corner of the video feed)

If there is no white border around the green dot and the word "LOCAL" does not appear next to "HQ" or "SQ", then the video is being relayed through the cloud.

Please note that security is maintained by using an encrypted TLS connection regardless of whether a live stream is accessed locally or relayed through the cloud. See the diagram below to observe how traffic flows directly between the accessing device and the camera:



Time synchronization

Verkada uses its own servers to sync the time on cameras over UDP 123. In addition to NTP, Verkada also uses NTS to secure the time synchronization process. Key exchange is done over TCP 4460. Organizations may also elect to use DHCP option 42 to sync to their own NTP server.

If you wish to change the time zone setting of a particular camera, you will need to change its address (this can be done from within the device settings page).

Firmware updates

Verkada cameras are equipped with a dual-partition firmware bank and can update their firmware over-the-air (OTA) without any intervention from the administrator. Firmware updates are pushed automatically. However, you can schedule firmware updates to happen during a pre-configured window.

When bringing Verkada cameras online for the first time, users can choose whether they would like to update the firmware immediately or delay the update. This makes it possible to configure and set up cameras without having to wait for each individual device to update.

To ensure failsafe updates, the camera will automatically revert to the previous version of the firmware if the update fails. Additionally, a random variable is introduced in the update process to prevent all cameras at a given location from rebooting at the same time.

Helpful tip

You can view the current camera firmware version within the device settings page. If the camera firmware is up to date, this will be denoted. If a newer firmware version is available, "Update Now" will appear below the current firmware version. Clicking "Update Now" will start the update process and override existing update schedules.

Firmware release notes provide insight into specific improvements and bug fixes and can be found here.

Configuring system alerts

Command admins can set up and subscribe to different types of device alerts, including:

1. **Online/Offline:** receive a notification any time a camera goes offline or comes back online. Learn how to set up camera status alerts here.
2. **Tamper:** receive a notification any time the onboard accelerometer detects someone or something making physical contact with the camera. Learn how to set up tamper alerts here.
3. **Occlusion:** receive a notification any time a camera's view is blocked or obscured. Learn how to set up occlusion alerts here.

Alert notifications can be delivered to Command users via text, email, and mobile app push notification. External contacts can also be designated to receive alert notifications via text (by inputting a valid phone number) and email (by inputting a valid email address). Verkada also boasts native integrations with Microsoft Teams and Slack, enabling alert notifications to be delivered directly to users within their workspaces.

If you are using a third-party system for ticketing/alerts, you can either use generic email addresses to direct emails to it, or utilize our API and webhook capabilities, as outlined here.

Helpful tip

Command users will only be able to receive alerts via text and email after the phone number and email address tied to their account have been verified. If you are unable to select these alert notification methods, navigate to the "Admin" page and select "My Account". A green check mark will appear next to the associated email address and phone number once they have been verified.

Note that online/offline alerts do not necessarily signal that the camera is no longer functioning, but rather that the communication with the cloud has been interrupted for a significant period of time. This can be triggered by events such as ISP outages, misconfigured firewall rules, or even routing issues. As long as the camera is still receiving power, it will continue to record. Relevant footage and information will be offloaded from the device once connection has been reestablished. If you want to get notified instantly that the camera, its cable, or the switch port they connect to has failed, please configure SNMP traps on the switch (or other alarming mechanisms provided by the vendor).