VIVOH

Vivoh Webinar Manager for Zoom

Installation and Administrator Guide

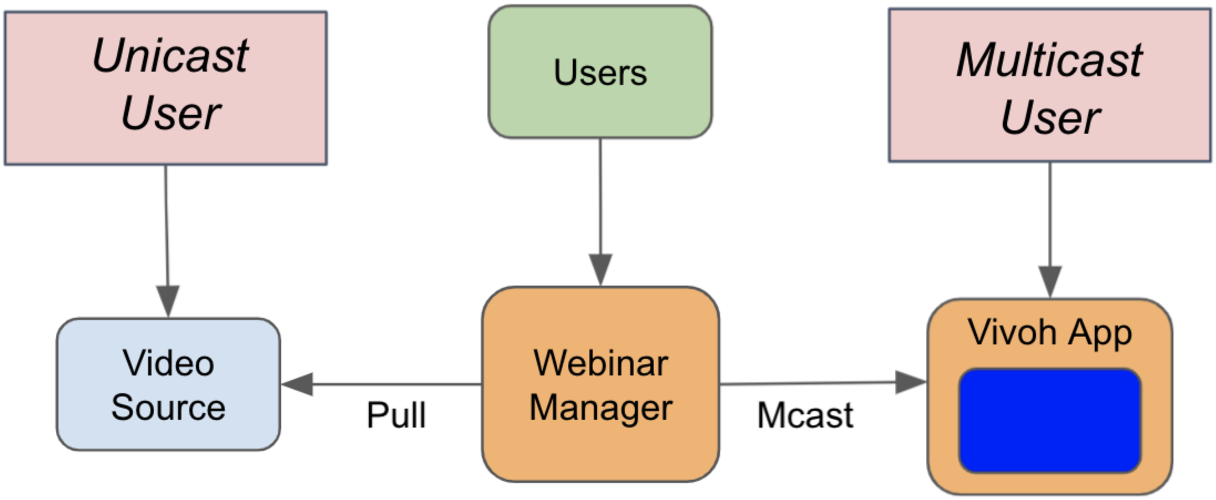
Contents:

1: Introducing the Vivoh Webinar Manager for Zoom

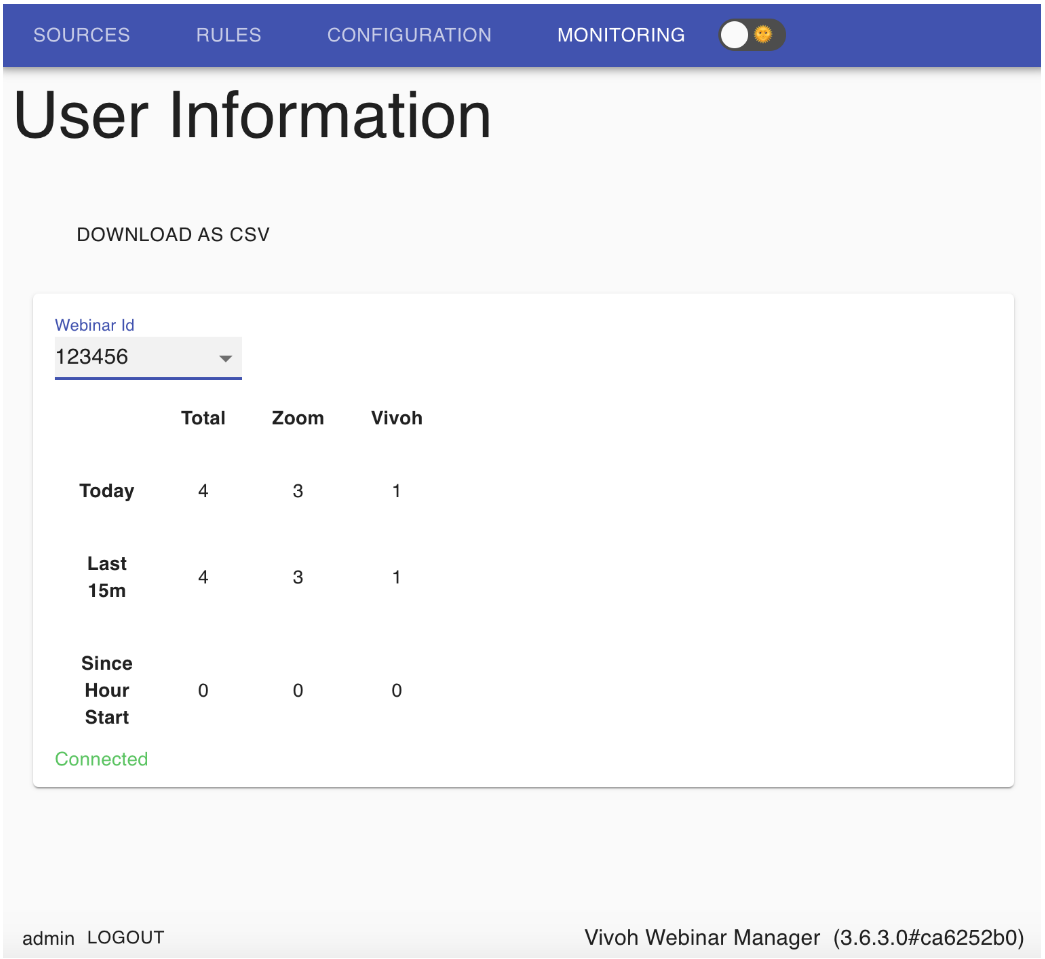
The Vivoh Webinar Manager is part of the Vivoh Zoom Multicast Solution which enables IT Service Delivery Managers to provide a seamless way to scale Zoom Webinars for enterprise webcasting using the multicast protocol. Please refer to the Vivoh Zoom Multicast Solution Technical Overview and Requirements for information about how the system works as a whole.

The Vivoh Webinar Manager enables IT Service Delivery Managers to register URLs that are used by the system, including a Multicast Address, a Stream Source, QA URL (event-specific third-party messaging application resource address, eg Sli.do), and a live captioning URL.

IT Managers then create Rules with the Vivoh Webinar Manager. Rules attempt to match the end-user webinar access requests based upon the client IP address (in the HTTP header sent by the user's browser) and the Webinar ID (part of the URL query string). If a match is successful then the Vivoh Webinar Manager will start an appropriate multicast encoder session (if not started already) and respond with HTML that will redirect the browser to a URL with the vivoh:// protocol. If the Vivoh Multicast App is installed then this will launch and join the multicast video stream. If no Rule match is found then the user is redirected to the appropriate Zoom Webinar URL or to the Webcast Fallback URL, if this is enabled.



The Vivoh Webinar Manager provides real-time statistics of how many users accessed the webinar via Zoom versus via the Vivoh App. This is aggregated and segmented by webinar and is accessible via the web interface. Administrators may download a CSV file with a current data snapshot. If configured, the Vivoh Webinar Manager will create a JSON-formatted log of each user request which also includes IP address and browser User Agent information.



User Information

Webinar Id: 123456

	Total	Zoom	Vivoh
Today	4	3	1
Last 15m	4	3	1
Since Hour Start	0	0	0

Connected

Vivoh Webinar Manager (3.6.3.0#ca6252b0)

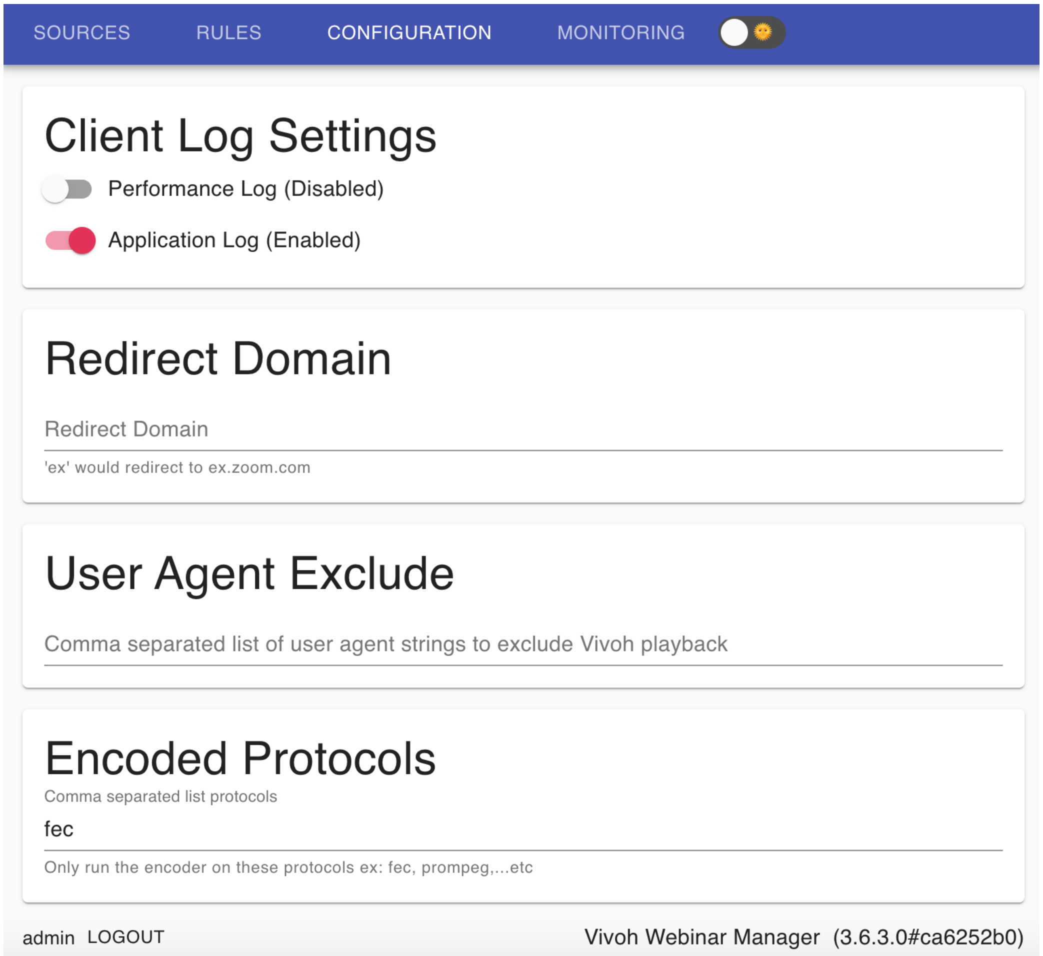
Vivoh Webinar Manager JSON Log Examples:

{"platform":"zoom","webinar":"12312312","destination":"vivoh","date":"2021-11-16T18:07:15.963Z","ip":"10.10.10.10","useragent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/95.0.4638.69 Safari/537.36"}

{"platform":"zoom","webinar":"12312312","destination":"zoom","date":"2021-11-16T18:07:46.295Z","ip":"::ffff:10.42.0.15","useragent":"Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_2) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/95.0.4638.69 Safari/537.36"}

Finally, the Vivoh Webinar Manager provides web-based configuration options. These include the ability to enable or disable two separate log files that can be generated locally for each user by the Vivoh Multicast App. One log is JSON-formatted and is intended to be pushed to a

Splunk Server via a local Splunk Forwarder application. The web-based configuration options include an option to route specific users, by User Agent, to Zoom in case it does not support multicast. They include an option to append a subdomain to the Zoom redirect URL for geographic routing and also include an option to specify which multicast streams are to be generated with the included Vivoh Multicast Encoder rather than set to be “passed through” as an already present multicast stream of the webinar.



Client Log Settings

○ Performance Log (Disabled)

● Application Log (Enabled)

Redirect Domain

Redirect Domain

'ex' would redirect to ex.zoom.com

User Agent Exclude

Comma separated list of user agent strings to exclude Vivoh playback

Encoded Protocols

Comma separated list protocols

fec

Only run the encoder on these protocols ex: fec, prompeg,...etc

admin LOGOUT

Vivoh Webinar Manager (3.6.3.0#ca6252b0)

The Vivoh Webinar Manager is a server application with a web-based management interface. All requests are via HTTPS, typically on port 443. It runs on all major operating systems. Optionally administrative access (also HTTPS) can be on another port, for example 8443.

2: Installation Prerequisites

Supported Browsers

To access the Vivoh Webinar Manager as a webinar participant or administrator, your computer must meet the following requirements:

- Windows
  - Internet Explorer 11 and above
  - Firefox 45.0.2 and above
  - Microsoft Edge 44 and above
  - Chrome 49 and above
- MacOS
  - Firefox 45.0.2 and above
  - Chrome 49 and above
  - Safari 10.1.2 and above
- Linux
  - Firefox 45.0.2 and above
  - Chrome 49 and above

Supported Operating Systems

The Vivoh Webinar Manager is supported by the following:

- Microsoft Server 2012 R2 or Microsoft Server 2016 and 2019
- RedHat Enterprise Linux 6.5, CentOS 6.5, Ubuntu 14.04 LTS and above
- Windows 10 and above

System Requirements

Memory: 8GB of RAM

Disk: 250MB

CPU: vCPU=4

3: Installing the Vivoh Webinar Manager

The Vivoh Webinar Manager for Windows is deployed with an installer that places all required files in a target directory. For Linux, the Vivoh Webinar Manager is distributed as a .deb archive that is installed with the ***dpkg*** command. The Vivoh Webinar Manager can be run manually, via command-line, or configured to run as a service. Skip to Virtual Appliance Installation below if you have been provided with an OVA image for the Vivoh Webinar Manager.

The Vivoh Webinar Manager installation consists of a binary and two configuration files, node_sqlite3.node and database.sqlite It also includes the Vivoh Multicast Encoder that is found in a subdirectory called “encoder”. Windows installation is performed via an interactive app.

To install on Linux, run the following: ***dpkg -i VIVOH_INSTALL_FILE.deb***. Start the service by running: ***systemctl start vivoh-webinar-manager***. Set Vivoh Webinar Manager to start as a service upon reboot by running: ***systemctl enable vivoh-webinar-manager***. You can see the status of the server with this command: ***systemctl status vivoh-webinar-manager***

You can run both the Windows and Linux versions with this command: ***vivoh-webinar-manager -a /etc/vivoh/vwm/vwm.ini*** (For Windows use your chosen directory) and this normally is run as administrator or root in order to allow it to bind to port 443.

Prior to starting the Vivoh Webinar Manager, administrators will want to edit the vmi.ini file which is located in the chosen folder during a Windows installation or /etc/vivoh/vwm/ on Linux. This file has three required settings and a number of optional settings.

The required settings are: ADMINISTRATOR, BROADCASTER, and PORT

For example, on Linux:

```
ADMINISTRATOR           = admin:!1Abc8888888,john:!1Abc888888E
BROADCASTER             = /usr/local/vivoh/encoder/bin/run.sh
PORT                    = 443
```

Please note that the ADMINISTRATOR setting must have at least one user:password combination where the password is at least 12 characters long with one special character, one capital letter, and one number. Otherwise, The Vivoh Webinar Manager will not start.

The optional settings are:

VERBOSE, ADMIN_PORT, NODE_ENV, VIVOH_TEMPLATE_PATH, JSON_USER_LOG, VIVOH_LIVE_WAIT_TOLERANCE, DATABASE_URL, DATABASE_LOGGING_ON, SESSION_SECRET, WEBCAST_LOGO_URL, WEBCAST_HEADER_COLOR, WEBCAST_FALLBACK_URL, VIVOH_DEBUG_USERNAME, REDIRECTOR, CERTIFICATES, IGNORE_FAILOVER_CERT_ERRORS, SKIP_SOURCE_VALIDATION, and FAILOVERS

Please consult Vivoh support for more information about the function of each of these settings, however the following sample settings may be sufficient for your purposes:

```
VERBOSE                             = 1
ADMIN_PORT                          = 8443
NODE_ENV                            = production
VIVOH_TEMPLATE_PATH                 = /etc/vivoh/vwm/foo.html
JSON_USER_LOG                       = /etc/vivoh/vwm/vwm_access.json
VIVOH_LIVE_WAIT_TOLERANCE           = 5
DATABASE_URL                        = mssql://root:root@localhost/vivoh
DATABASE_LOGGING_ON                 = true
SESSION_SECRET                      = “mySecret2021”
WEBCAST_LOGO_URL                    = “https://company.com/images/vivoh.png”
WEBCAST_HEADER_COLOR                = “white”
WEBCAST_FALLBACK_URL                = “https://company.com/event/${webinar_id}”
VIVOH_DEBUG_USERNAME                = “vivoh_debug”
REDIRECTOR                          = false
CERTIFICATES                        =
/etc/vivoh/privkey.pem,/etc/vivoh/cert.pem,/etc/vivoh/chain.pem
IGNORE_FAILOVER_CERT_ERRORS         = false
SKIP_SOURCE_VALIDATION              = true
FAILOVERS                           = “https://vivoh_nyc.customer.com”
```

Virtual Appliance Installation:

The Vivoh Webinar Manager can be deployed from a provided OVA image. This image has been tested with VMWare ESXi and is compatible with versions 6.0 or later. This image will create 2 CPU sockets with 4 cores each, reserve 4GB of RAM, and allocate 40 GB of Hard Drive space. See a screenshot image of a successful OVA deployment settings screen at the end of this document.

Once the OVA is loaded into its host server, it will start with DHCP and SSH enabled for remote access. Alternatively, administrators can use the “Launch Web Console” to gain access via a command line interface. This is typically done to configure static networking and to enable the firewall.

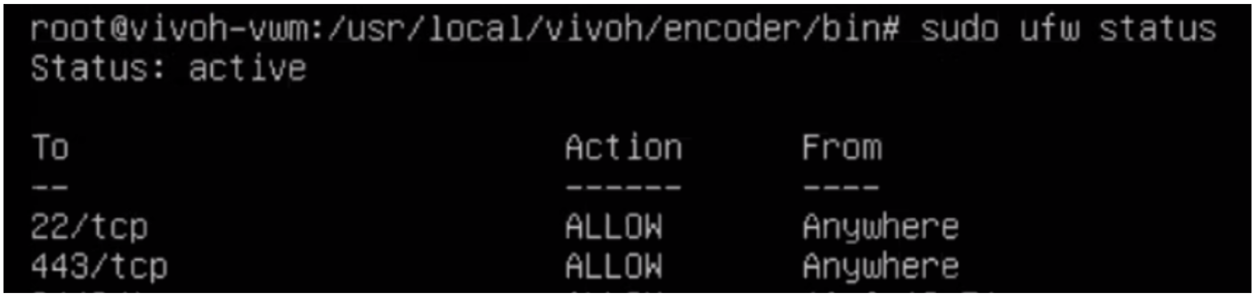
Vivoh software is located in /usr/local/vivoh/ and the configuration file is /etc/vivoh/vwm/vwm.ini. This includes the Vivoh Multicast Encoder which is in /usr/local/vivoh/encoder.

Networking is configured with Netplan. Run ***ip addr*** to locate the adapter name then edit /etc/netplan/01-netcfg.yaml (example below) then run ***sudo netplan apply***

Example Netplan YAML configuration file:

```
network:
  ethernets:
    ens3:
      dhcp4: false
      addresses:
        - 192.168.121.221/24
      gateway4: 192.168.121.1
      nameservers:
          addresses: [8.8.8.8, 1.1.1.1]
version: 2
```

The firewall is configured with the ***ufw*** command. Type ***sudo ufw status*** to see the current firewall status. Then ***sudo ufw allow ssh***, ***sudo ufw allow https***, and ***sudo ufw enable***



```
root@vivoh-vwm:/usr/local/vivoh/encoder/bin# sudo ufw status
Status: active

To                         Action      From
--                         ------      ----
22/tcp                     ALLOW       Anywhere
443/tcp                    ALLOW       Anywhere
```

A best practice is to enable the separate administrator port of 8443 for the Vivoh Webinar Manager and, optionally, only allow access to this service from specified IP ranges.

To limit access to this port by specified IP ranges, type (for example): ***sudo ufw allow proto tcp from 192.168.1.100 to any port 8443***. Run ***sudo ufw status*** to confirm these settings.

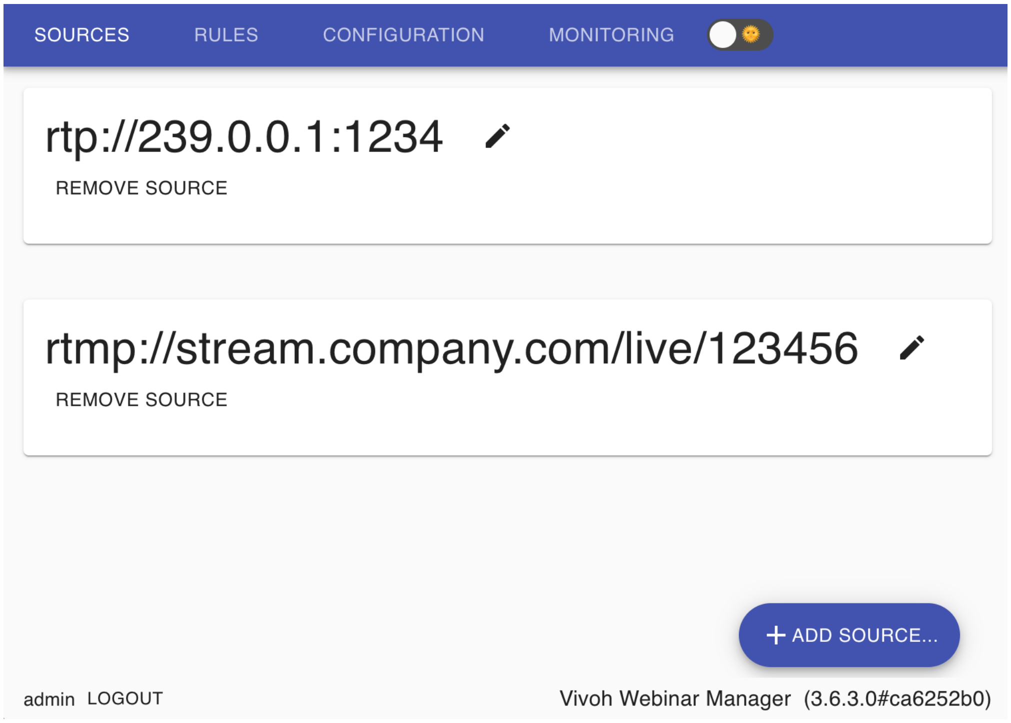
Please note that the Vivoh Webinar Manager has several features to prevent intrusion, including:

1. Brute Force Spoiler
2. Session Timeout (30 minutes of no activity and 3 hours of any activity)
3. Random Server-side Session Secret (Override with the SESSION_SECRET setting to enable persistent authentication across a cluster of servers, if desired.)

4: Working with the Vivoh Webinar Manager

The Vivoh Webinar Manager is part of the Vivoh Zoom Multicast Solution which enables IT Service Delivery Managers to provide a seamless way to scale Zoom Webinars for enterprise webcasting using the multicast protocol. The Vivoh Webinar Manager is the server that handles the initial user request for the webinar (regardless of how they will view it). This server contains Rules and URLs that direct the user to either the Zoom Webinar App, the Vivoh Multicast App, or to a designated unicast fall back site as specified by the WEBCAST_FALLBACK_URL option.

IT Managers will initially configure the Sources to include available multicast Group Addresses (plus ports), unicast video sources (including an optional variable for ${webinar_id} that enables dynamic sources based upon each unique Webinar ID), QA URLs, and captioning URLs.

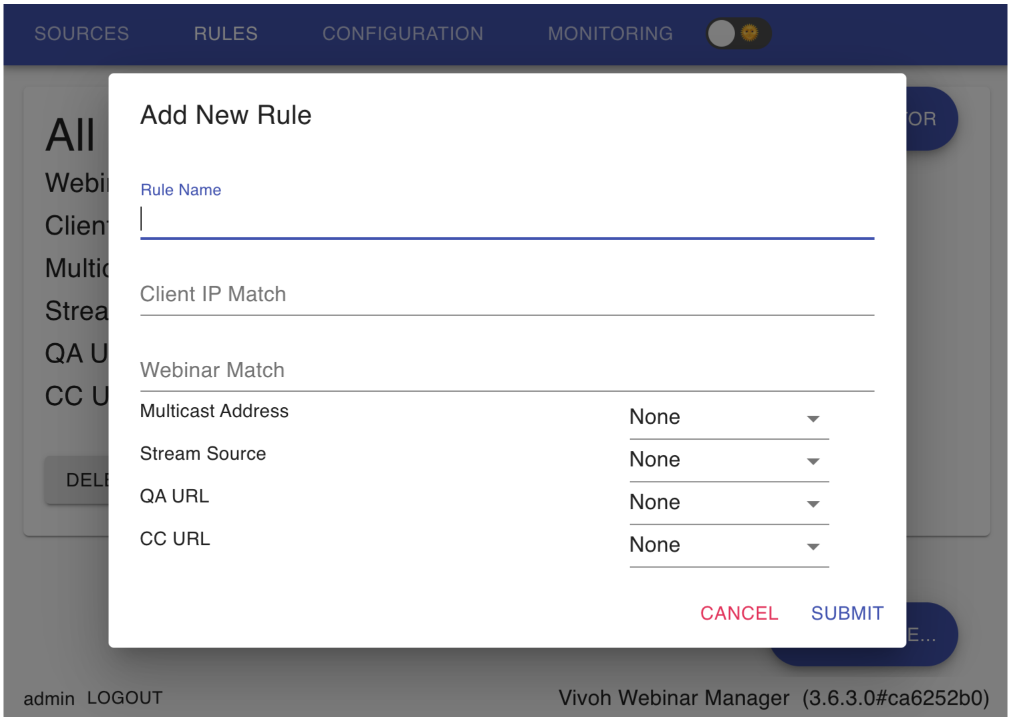


rtp://239.0.0.1:1234

rtmp://stream.company.com/live/123456

Vivoh Webinar Manager (3.6.3.0#ca6252b0)

IT Managers will then create Rules that match sets of potential IP addresses of clients as well as Webinar IDs. CIDR notation and wildcards (“*”) are supported for IP addresses and multiple entries may be added as a comma-delimited list. Webinar ID matches may be exact, a catch-all wildcard value (“*”), or a regular expression (eg. “^212” to match any Zoom Personal Meeting Room ID that starts with 212).



Add New Rule

Rule Name

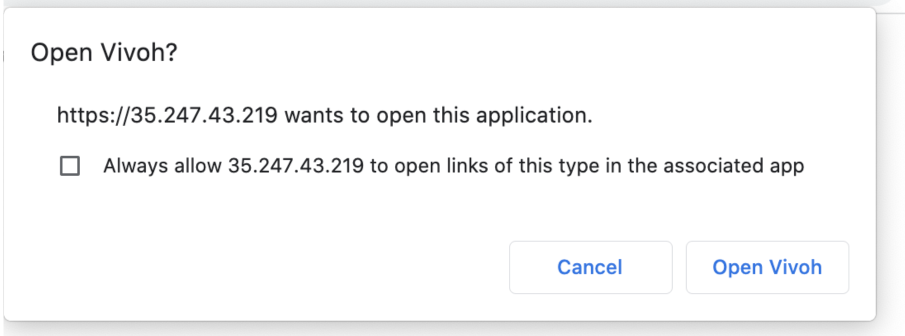
Client IP Match

Webinar Match

Multicast Address	None
Stream Source	None
QA URL	None
CC URL	None

Vivoh Webinar Manager (3.6.3.0#ca6252b0)

The Vivoh Webinar Manager will try to find a match for each client webinar join request and then start the Vivoh Multicast Encoder (if not already started) and redirect the user to the Vivoh Multicast App for each successful match. Otherwise, they will be redirected to the corresponding Zoom Webinar URL or Webcast Fallback URL depending upon configuration options.

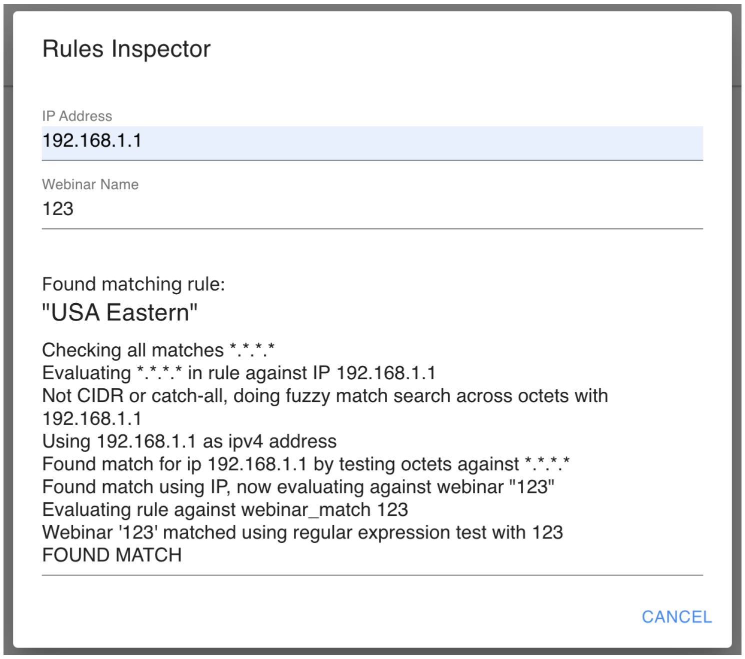


Open Vivoh?

https://35.247.43.219 wants to open this application.

☐ Always allow 35.247.43.219 to open links of this type in the associated app

The Rules Inspector enables administrators to try example user IP address and Webinar ID values to confirm that they are matching according to expectations.



Rules Inspector

IP Address
192.168.1.1

Webinar Name
123

Found matching rule:
"USA Eastern"

```
Checking all matches *.*.*.*
Evaluating *.*.*.* in rule against IP 192.168.1.1
Not CIDR or catch-all, doing fuzzy match search across octets with
192.168.1.1
Using 192.168.1.1 as ipv4 address
Found match for ip 192.168.1.1 by testing octets against *.*.*.*
Found match using IP, now evaluating against webinar "123"
Evaluating rule against webinar_match 123
Webinar '123' matched using regular expression test with 123
FOUND MATCH
```

As documented in the Vivoh Zoom Multicast Solution Technical Overview and Requirements, Zoom Hosts will use the Custom Live Streaming URL option to push video via RTMPS to the Vivoh Media Server. They should use the Webinar ID as their “Stream Key” as Vivoh will redirect the user to the Vivoh App or back to the Zoom App with the same Webinar ID.

Instead of using the Zoom webinar link, users are given a link to the Vivoh Webinar Manager which will redirect them to either the Vivoh Multicast App or the standard Zoom Webinar App depending upon their IP address and corresponding matching rules in the Webinar Manager. An example of this link is: https://vivoh.company.com/zoom/123456789 where 123456789 is the Webinar ID. Use the ${webinar_id} variable for the source to match the “Stream Key”.

Advanced deployments may require IT Managers to install multiple Vivoh Webinar Manager servers. For example, they may deploy on public IP addresses for the purpose of handling initial user requests for redirection to internal Vivoh Webinar Managers or Zoom Webinar URLs, as needed. This allows the use of one webinar link for all potential users. To implement this feature, set the REDIRECTOR property to ***true*** and create a Rule with a Stream Source of the internal Vivoh Webinar Manager URL (use Anycast DNS or F5 BIG IP to load balance servers) and the external user request will be redirected to the internal Vivoh Webinar Manager.

Advanced deployments may require redundancy and centralized configuration management. The Vivoh Webinar Manager handles user requests statelessly and multiple managers can be configured to read/write from a centralized configuration database. Redundant multicast sessions can be automatically initiated by using the FAILOVERS property to configure them to start both a local encoder and a remote encoder. The Vivoh Multicast App will failover between them and keep trying until one multicast source resumes broadcasting.

5: Working with Logs and Implementing Analytics

The Vivoh Webinar Manager can be configured to log user requests in the JSON format to a log file that is specified by the property: JSON_USER_LOG (see configuration example above). This will log the IP Address, User Agent, Webinar ID, and whether they were routed to the Vivoh App or the Zoom App.

In addition to server-side logging, the Vivoh Webinar Manager enables real-time logging for analytics via a local application log and a performance log which are stored on each user's home directory. Vivoh recommends Splunk for analytics processing and the use of Splunk Forwarder agents to pick up the performance log files that are generated by the Vivoh Multicast App and push these periodically to the central Splunk server. Reports are available via the Splunk interface. The Vivoh App logs to the user's home directory as 'Vivoh-App.log' and 'Vivoh-App.json'. Vivoh-App.json contains an entry each minute with essential analytics data.

IT Service Delivery Managers can enable and disable each of the log files for all users that access the webinar via the specified Webinar Manager via its Configuration menu.

In addition to these logs, the Vivoh Webinar Manager will push real-time encoder information, including video and audio packet counts to the system log. You can tail this on Linux by typing: ***tail -f /var/log/syslog*** and some organizations may choose to push these logs to Splunk as well. An example of this data is shown below:

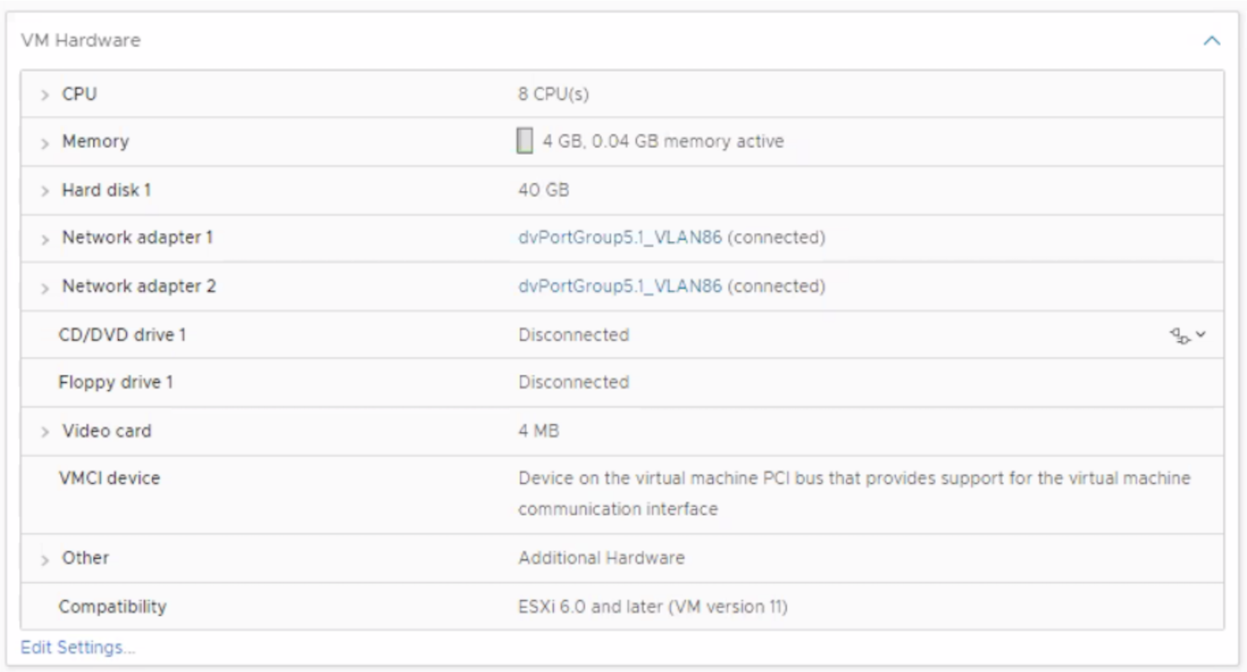
Packet Summary: Video RTP:118910362, Video RTCP:2000, Audio RTP:3639279, Audio RTCP:1920, Total:122553561

6: Working with Messaging

The Vivoh Webinar Manager enables IT Service Delivery Managers to provide webinar hosts with a way to bring their audience members into existing corporate communications systems for messaging during their live events.

The Vivoh Multicast App has an optional QA button which is automatically configured for each event by specifying the QA URL in the appropriate Vivoh Webinar Manager Rule. IT Service Delivery Managers will configure the approved base URL for the internal messaging service for each channel or for specific webinars, and Hosts can set up "rooms" based upon the Webinar ID. For example, https://messages.company.com/rooms/${webinar_id} where ${webinar_id} will match the Webinar ID of the Zoom Webinar. This link will open when the user clicks on the QA icon in the App.

8: Vivoh Webinar Manager Images

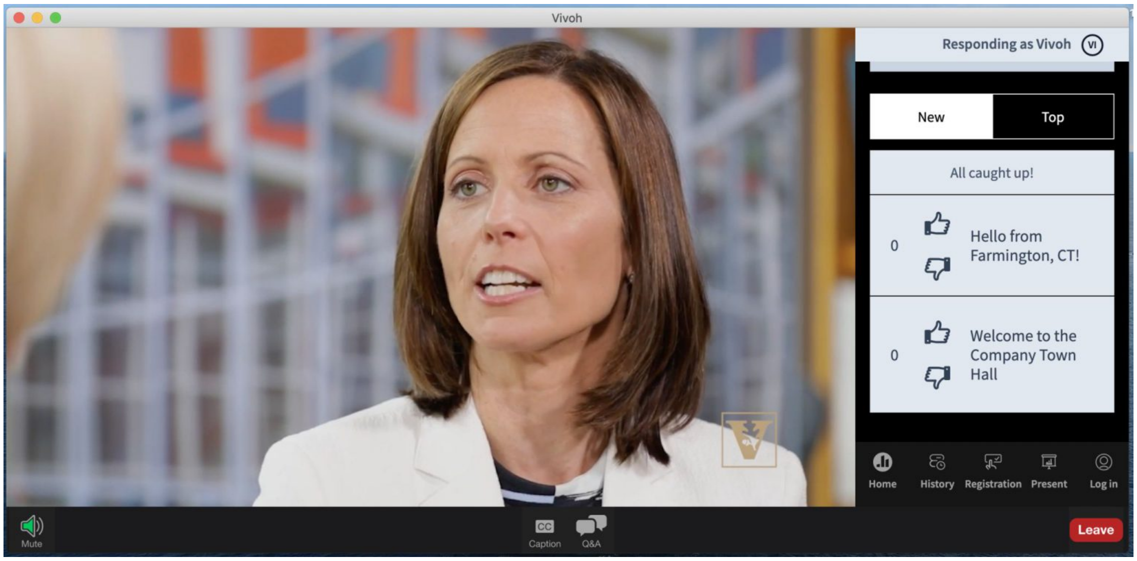


VM Hardware

CPU	8 CPU(s)
Memory	4 GB, 0.04 GB memory active
Hard disk 1	40 GB
Network adapter 1	dvPortGroup5.1_VLAN86 (connected)
Network adapter 2	dvPortGroup5.1_VLAN86 (connected)
CD/DVD drive 1	Disconnected
Floppy drive 1	Disconnected
Video card	4 MB
VMCI device	Device on the virtual machine PCI bus that provides support for the virtual machine communication interface
Other	Additional Hardware
Compatibility	ESXi 6.0 and later (VM version 11)

Edit Settings...

Figure 1: VMWare OVA Settings Screen



Responding as Vivoh

All caught up!

0 Hello from Farmington, CT!

0 Welcome to the Company Town Hall

Figure 2. Vivoh Multicast App with Poll Everywhere QA Messaging