How to debug, troubleshoot and monitor VoIP using MikroTik
Whoami
- Voicenter
- Homer

Voip Protocols
- SIP
- RTP
- RTCP

Mikrotik VoIP Setup
- QOS
- Provisioning
- Monitoring

Tools for VoIP Analytics
- Wireshark
- Sngrep
- CaptAgent
- RtpAgent

Homer Cloud
- Troubleshooting
- Monitoring
- Alerting
Shlomi Gutman

CTO of Voicenter (Israel)
VP of Cloud Products at QXIP (Amsterdam)
Hi.

Shlomi Gutman.
Founder and CTO at Voicenter.
Open-Source Telephony expert.
Built my first computer when I was 8 years old.

shlomi@voicenter.com
Voicenter is a leading telecommunication technology company providing top-tier business telephony since 2007.

We are delivering a ‘One-stop-shop’ solution for business all around the world.
Voicenter – Cloud Contact Center

- Cloud-based Phone system
- Hybrid Solution
- Real Time Dashboards
- Workforce Management
- Dialers
- Api
- Integration
Voicenter – Cloud Contact Center

**Voicenter** is an Israel based business providing a solid array of services to its customers, including:

- Cloud based Phone Systems
- Hybrid Solutions
- Real Time Dashboards
- Workforce Management
- Dialers
- Api
- Integration

---

**QXIP BV** is an Amsterdam based R&D Company specializing in *Open-Source* and *Commercial* Voice Technologies deployed and trusted by thousands of businesses worldwide, include large telephony and network operators, voice service carriers, voip service providers, cloud service providers, call center operators and voice equipment vendors.

**QXIP** Capture Technologies are natively implemented in all major OSS voip platforms such as *Kamailio*, *OpenSIPS*, *FreeSWITCH*, *Asterisk*, *RTPEngine* and many tools such as *sipgrep*, *sngrep*.
Elephant in the server room
VoIP & RTC Problems

- Connectivity Problems
- Call Quality Problems
- Security Problems
- Multi Equipment management
- Hard to troubleshoot
- Mission Critical Application
VoIP Protocols

- SIP / WEBRTC / TLS – Signaling Protocols
- RTP - Media Protocol
- RTCP – Real Time Control Protocol
SIP Flows - Basic

User A

"Calls"
18.18.2.4

INVITE: sip:18.18.2.4

180 - Ringing

200 - OK

RTP

Talking

Hangs up

User B

Rings

Answers

ACK

Talking

RTP

200 - OK

BYE

200 - OK
RTP - How Digital Audio Works

00111 01000 01001 01001 01000 00101 10110 11000 11001
11001 11000 10111 10100 10001 00010 00111 01001 01010
01001 00111 00000 11000 11010 11010
11001 11000 10110 10001
RTCP-RTP (Quality) Control Protocol

```
"event": {
    "media": "audio",
    "base": 48000,
    "lsr": 37971368,
    "lost": 0,
    "lost-by-remote": 0,
    "jitter-local": 18940,
    "jitter-remote": 0,
    "packets-received": 39,
    "packets-sent": 40,
    "bytes-received": 6708,
    "bytes-sent": 7280
}
```

1 RTCP packet per RTP stream each 5-10 seconds
How Can Mikrotik Push my Voip Packets?
Traffic Shaping Concept

Line Limit
Real Life Line Limit
None VoIP Limit

VoIP Dedicated Bandwidth

without Traffic Shaping
with Traffic Shaping

Losers make promises they often break.
Winners make commitments they always keep.
VoIP QoS Best Practice

• Address List Maintenance
• Connection Marking
• Packet Marking
• Queues configuring
Media servers Import Script

```plaintext
{do {
/tool fetch url="http://mikrotik.XXXX.com/script/MediaServer.rsc" mode=http
} on-error={:put "Error on downloading OF MediaServer Script --> http://mikrotik.XXXX.com/script/IPList/MediaServer.rsc"};
#Start Loading
:do {
/import file-name=MediaServer.rsc
} on-error={:put "Error on Running MediaServer.rsc"};
}
```
Media servers HTTP Response

:put " * *****Add XX.XX.XX.XX MGW 01 to MediaServer**************"
:do {
/ip firewall address-list add address=XX.XX.XX.XX list=MediaServerList comment="MGW 01 ->MediaServer"
} on-error={ :put " Failed to add XX.XX.XX.XX MGW 01 to MediaServer probably already there "};

:put " * *****Add XX.XX.XX.XX MGW 01 to MediaServer**************"
:do {
/ip firewall address-list add address=XX.XX.XX.XX list=MediaServerList comment="MGW 01 ->MediaServer"
} on-error={ :put " Failed to add XX.XX.XX.XX MGW 01 to MediaServer probably already there "};
Connection Marking

New Mangle Rule

General
- Chain: prerouting
- Src. Address:
- Dst. Address:
- Protocol: udp
- Src. Port:
- Dst. Port: 5060, 5061, 10000-32767
- Any. Port:
- P2P:
- In. Interface:
- Out. Interface:
- In. Interface List:
- Out. Interface List:
- Packet Mark:
- Connection Mark:
- Routing Mark:
- Routing Table:
- Connection Type:
- Connection State: [new, invalid, established, related]

Advanced
- Src. Address List:
- Dst. Address List: MediaServerList
- Layer7 Protocol:
- Contact:

Action:
- Action: mark connection
- Log: [Log]
- Log Prefix: [VoipTrafficConnectionMark]
- Passthrough [on, off]

Statistics

OK
- Cancel
- Apply
- Disable
- Comment
- Copy
- Remove
- Reset Counters
Pocket Marking
Queue for non VoIP traffic
VoIP Monitoring Using Mikrotik

Switch Layer mirroring
  - Good Performance

Packet Sniffer stream
  - Bad Performance

Pcap File analytics
  - Ugly from any perspective
Packet Sniffer Setup
(TZSP - Packet Sniffer encapsulation)
RTP in Wireshark

Max delta = 94.99 ms at packet no. 974
Max jitter = 5.09 ms. Mean jitter = 0.70 ms.
Max skew = 191.64 ms.
Total RTP packets = 1047 (expected 1047) Lost RTP packets = 0 (0.00%) Sequence errors = 0
Duration 25.23 s (-731 ms clock drift, corresponding to 7768 Hz (-2.90%)
SIP + RTP in HOMER Cloud
CaptAgent / RTPAgent

modular capture agent supporting multiple sockets, protocols and transport methods

- **TZSP** support for MicroTik Packet Sniffer encapsulation
- **HEP** support for **HOMER** Cloud Analytics and Monitoring
VM: Debian 8
- RTPAgent w/ TZSP socket
The **CALL SEARCH** functionality is one of the most used tools to locate, analyze and extract present and past call sessions.

The Call Search tool obeys the global **TIME-RANGE** as its primary filter, extended by a customizable number of user defined parameters targeting session headers and parameters (more in the next slide).

The **SEARCH** functionality also offers programmable “Search Profiles” per group used to automatically include and match multiple dialing patterns (international/national) prefixes (00/+ or routing prefixes from a bare number with no additional user interaction required.

To maximize the platform’s full potential, a “**two-tier**” approach is also possible and suggested, with a first initial group of results returned by the backend and complex filtering by any field can be performed client-side using advanced regex patterns and wildcard matching.

**NEXT: TRACKING CALLS AND SESSIONS**
Search Results for Calls will be returned in a table, ordered by timestamp and ready to use.

PCAPTURE core is session aware and can display call status in realtime and aggregate all messages, statistics and logs in a single object, with automatic correlation to any other connected B2BUA legs.

To investigate Session Details, just click on a result.

Search Results table columns can be configured based on user preference to show or conceal any of the available session and protocol parameters.
**UI: Tracking Calls and Sessions Details**

(continued)

Session Details will be returned when selecting one or more result rows. The API will automatically fetch all correlated call data in the current Time Range.

The “Session Detail” window features several Tabs presenting available correlated information about the Session (or Session Group) being displayed.

**Call Session** tab presents packets in *Shark-View* mode

Each packet and message can be inspected in any display mode by simply clicking the corresponding row or object to reveal the full original payload data.
UI: Tracking Calls and Sessions Details

(continued)

Sessions involving several devices with hops traversing multiple systems can quickly get complex.

PCAPTURE’s Call Flow tool automatically correlates hosts and messages and presents them in an easy-to-interpret format well familiar to voice experts of all seasons and capable of handling unlimited legs.

Call Flow tab presents packets in Signaling Flow mode.

Each packet and message can be inspected in any display mode by simply clicking the corresponding row or object to reveal the full original payload data.
UI: Tracking Calls and Sessions Details

(continued)

When additional data about the Session being inspected is available, PCAPTURE will automatically present it to the end-user without any interaction.

The **Export Tab** provides dynamic methods to Save, Archive or Share the current set in different formats.

- **Voice Quality** tab presents stream RTP-RTCP quality reports
- **Geo Maps** tab presents the approx. IP Geolocation of User Agents
- **Devices** tab presents details about *Registered* SIP User-Agents
UI: Tracking Media Session Quality

(continued)

The Voice Quality tab presents metrics related to media sessions as reported by User-Agents, Passive Network Probes and Media Control Protocols, providing correlated data useful when analyzing complex RTP media paths between SIP Endpoints.

User-Agent generated media reports (RTCP-XR, X-RTP-Stat, P-RTP)

CAs can automatically report for monitored streams at variable or static rates, with each report carrying all RTP metrics and MOS

RTP Report from passive network analysis with granular metrics

CAs can also capture and aggregate RTCP control protocol messages and extract metrics and statistics to calculate a MOS

RTCP Report from analysis and aggregation of User-Agent reports
UI: Tracking Registrations and Devices

**PCAPTURE** features a dedicated tool for **Searching** and **Filtering** registration with **Expiration tracking**, integrated with Call Search tools:
CDR SEARCH & FILTER:

**PCAP: CDR Count**

- **257** CDRs

**PCAP: CDR Cost Metrics**

- **$122** Total CDR Cost
- **$9** Highest Rated CDR
- **$0.48** Avg. CDR Cost

**PCAP: Session Status**

- **FINISHED**: 132
- **CANCELED**: 77
- **USER FAILURE**: 43
- **USER_BUSY**: 3
- **DECLINE**: 2

**PCAP: CDR Search Extended**

<table>
<thead>
<tr>
<th>Time</th>
<th>callid</th>
<th>from_user</th>
<th>from_domain</th>
<th>to_user</th>
<th>to_domain</th>
<th>duration</th>
<th>status_text</th>
<th>calls</th>
<th>d_prefix</th>
<th>d_total_cost</th>
</tr>
</thead>
<tbody>
<tr>
<td>October 31st 2016, 18:15:11.000</td>
<td>14738729298-45776256449</td>
<td>05446</td>
<td>172.3</td>
<td>073</td>
<td>073</td>
<td>00.02</td>
<td>FINISHED</td>
<td>4.4</td>
<td>972</td>
<td>0.04</td>
</tr>
<tr>
<td>October 31st 2016, 18:15:14.000</td>
<td>9b7f14e531070a-0002a</td>
<td>423</td>
<td>212.199.1</td>
<td>073</td>
<td>073</td>
<td>00.13</td>
<td>FINISHED</td>
<td>0</td>
<td>1</td>
<td>0.195</td>
</tr>
<tr>
<td>October 31st 2016, 18:15:11.000</td>
<td>5f517f66479861151</td>
<td>405</td>
<td>212.199.1</td>
<td>073</td>
<td>073</td>
<td>00.13</td>
<td>FINISHED</td>
<td>0</td>
<td>1</td>
<td>0.195</td>
</tr>
<tr>
<td>October 31st 2016, 18:15:11.000</td>
<td>795e57929189848</td>
<td>405</td>
<td>212.199.1</td>
<td>073</td>
<td>073</td>
<td>00.00</td>
<td>CANCELED</td>
<td>0</td>
<td>972</td>
<td>0</td>
</tr>
<tr>
<td>October 31st 2016, 18:15:11.000</td>
<td>212.199.1</td>
<td>073</td>
<td>073</td>
<td>073</td>
<td>073</td>
<td>00.00</td>
<td>CANCELED</td>
<td>0</td>
<td>972</td>
<td>0</td>
</tr>
<tr>
<td>October 31st 2016, 18:15:11.000</td>
<td>212.199.1</td>
<td>073</td>
<td>073</td>
<td>073</td>
<td>073</td>
<td>00.00</td>
<td>CANCELED</td>
<td>0</td>
<td>972</td>
<td>0</td>
</tr>
<tr>
<td>October 31st 2016, 18:15:11.000</td>
<td>212.199.1</td>
<td>073</td>
<td>073</td>
<td>073</td>
<td>073</td>
<td>00.00</td>
<td>CANCELED</td>
<td>0</td>
<td>972</td>
<td>0</td>
</tr>
</tbody>
</table>
Alerting and Fraud detection

**SA: Time Series Visualization in Kibana**

Complex reports can be created leveraging all available metrics and time series, including comparisons across different data ranges: