Edge Server Based Call Recording for Microsoft Lync

Józsefné Kovács[1], Péter Krasznai[2] and Norbert Kakas[1]
1: Óbuda University, Alba Regia University Center, H-8000 Székesfehérvár Budai u. 45, Hungary
2: Kovax Kft., H-1115 Budapest Petzval Jozsef u. 6, Hungary
kovacs.jozsefne@arek.uni-obuda.hu, krasznai@verba.com, norbicsek@gmail.com

Abstract **- Microsoft unified communications solutions are quickly gaining adoption worldwide. Microsoft entered the instant messaging and collaboration market with Live Communication Server in 2003 that later evolved into Office Communications Server and finally Lync in 2010. These systems are not designed to support standards based call recording, therefore lawful interception and compliance recording is cumbersome. Previously only conference-based and passive recording technologies were available that either slowed down the call setup process and rendered the telephony system unreliable or required detailed knowledge of the underlying network infrastructure. To circumvent the inconvenience of these approaches Verba Technologies have developed a centralized recording solution.**

I. INTRODUCTION

The growing use of VoIP (Voice over Internet Protocol) has made the recording of these systems a major question due to multiple reasons eg. compliance with law, security causes, evaluation of agents at call centers and data verification. The VoIP calls' specialty is that the voice travels via computer networks and that allows to record it by listening into these systems through a SPAN (Switched Port Analyzer) port. There are two existing methods to record a VoIP call. The first one is that the VoIP Gateway is spanned and in and outbound traffic is returned, both RTP (Real-time Transport Protocol) and call signaling. This method is useful for large companies with separated departments. However it has a drawback that this solution is unable to record the internal calls. Another solution is that, every phone in the voice system is mirrored, which grants the ability to record internal calls but this method can have a very difficult architecture in a large company's network and phone system. Another drawback of this solution is if a company uses encrypted call flows it is impossible to record, because the necessary encryption keys are missing. The previously discussed ways are „passive recording" solutions. For larger network there is another method that is called centralized recording.

II. CENTRALIZED CALL RECORDING

Centralized VoIP recording is a concept that can be realized in multiple ways:

- **PBX-based** - The PBX offers a call recording interface. This interface can be proprietary (Cisco Call forking, CUCISA, Avaya media forking...) or the latest common standard based (SIPREC). Currently SIPREC is new at the market and vendors are only in planning phase of supporting it.
- **MTP-based** - Calls are terminated on an MTP (Media Termination Point), traffic of MTP is recorded. MTP can be Session Border Controller or RTP proxy or transcoder resource in the network. Applicability depends on the deployment and system capabilities.
- **Conference-based** - Call participants are joined to a conference with the recorder. To escalate the conference PBX APIs can be used.

III. MICROSOFT LYNC

Microsoft Lync is an enterprise unified communications platform. Lync software has a big opportunity in enterprise voice networks. Lync users are able to send instant messages to each other, make a phone call or start audio and video conferences and users are able to share their applications and desktops. Lync has an Office integration which makes it easier to work with and this is the reason why it has a rapidly gaining user base.
A Microsoft Lync system contains several type of server roles:

- **Front End -** The Front-end server is responsible for the client registration and call routing requests for clients. Front-End server contains all the configuring features such as Control Panel, Address Book.

- **Back End -** This role is the SQL database and in most cases, it is on the same server as the Front-end.
- **Edge -** The Edge role is responsible for the remote capabilities of the Lync platform such as federation with other departments, remote access and public internet connectivity (Windows Live, Yahoo!).
- **Director** - The director role performs authentication and provides a stop-off point for all external traffic.
- **Mediation -** This is the interface to the Public Switched Telephone Network (PSTN). It's working together with a media gateway device via standard SIP call controlling protocol.
- **Monitoring -** It allows to create detailed SQL reports.
- **Archiving** - The Archiving role provides capture of all IM and store these.
- **Audio/Video Conferencing:** This role is responsible to join participants into a conference call. Since it has audio and video processing tasks, it is recommended to install on a dedicated server.
- **SBA -** The SBA (Survivable Branch Appliance) is an all-in-one device which provides some of the functionality of a pool (registrar and routing).

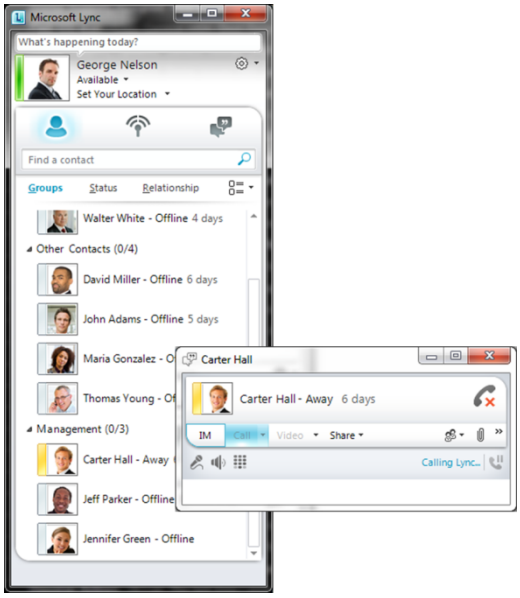


Figure 1. Microsoft Lync client

Until now Microsoft Lync systems could be recorded by the mentioned two passive recording methods, but from now on Verba Technologies can provide a centralized Edge based recording solution.

IV. VERBA RECORDING SOLUTION

A. Calls in a Lync environment

Call signaling flows always involve Front-Ends servers. Since Lync uses the ICE protocol for media channel establishment, it always tries to select the optimal media path:

- **two Lync clients are on the same LAN** - peer to peer media flow
- **one of the clients is outside of the corporate network** - direct media flow via a NAT gateway or relayed flow via Edge servers
- **both clients in the call are outside of the corporate network** - media does not flow through the corporate network (this makes non-centralized passive recording impossible)

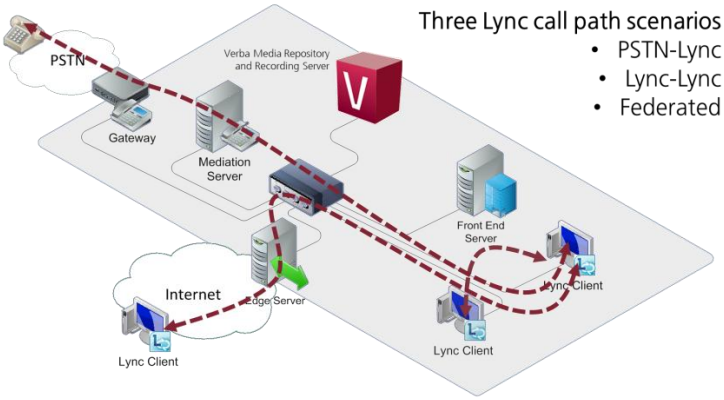


Figure 2. Calls in a Lync environment

B. Fundamentals of the Verba recording solution

The Verba Lync recording solution is based on passive recording enhanced integration of an MSPL filter to provide call signaling (SIP) messages for the recorder. The using of MSPL filter is important because the original call flow and the call signaling are encrypted.

The Verba recording system executes the followings for each recorded call:

1. MSPL filter is installed on each Front End and captures signaling
2. MSPL filter sends SIP signaling to the recorder on a secure TLS connection
3. Recorder parses signaling, extracts call metadata, media stream information
4. Recorder maps SRTP streams for calls
5. Encrypted media (SRTP streams) is captured from the network
6. Recorder decrypts and decodes media

7. Recorder mix/interleaves audio channels and transcodes to a configured audio output format on the fly

C. Using the Lync Edge server for call recording

Forcing media flows via Edge servers

In order to force the media to traverse the Edge Server, Verba developers manipulate the ICE offer/answer messages to use the relay candidate.

Verba developers believed that there are two ways to achieve this:

- Prevent connection through candidates
- Changing candidates priority to prefer the relay candidate: we have not tried this approach, but we are afraid that via dynamic candidate discovery the agents are able to learn the original priority for peer-reflexive candidates which can result in direct media path

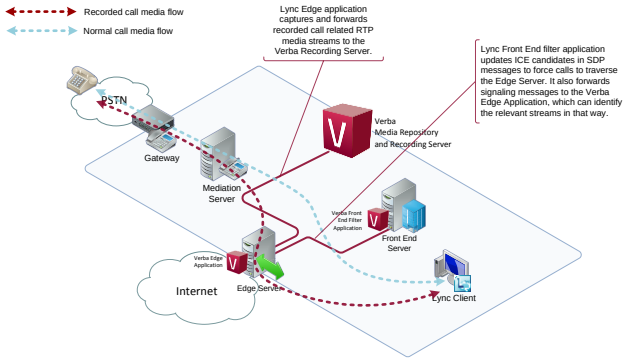
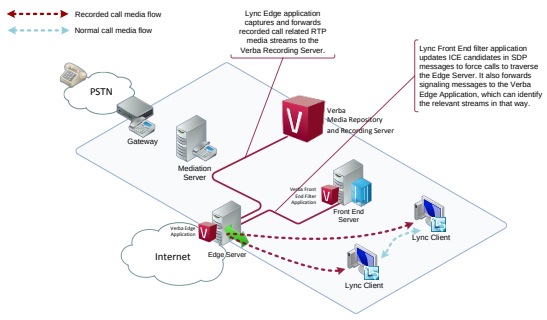


Figure 3. A Lync to PSTN call

Prevent connection through candidates

In this approach, we change the port number for each unwanted candidate to an invalid one. To have a minimal impact on the service, we only update the candidates if all the following conditions are satisfied:

- The call is to be recorded
- The current stream is to be recorded (audio or video)
- Relay candidates are present
- Candidate type is not relay



D. ICE (Interactive Connectivity Establishment)

All VoIP protocols (H.323, SCCP, SIP, XMPP) advertise media receiving socket addresses during call setup/media negotiation. This approach leads to a problem when NAT (network address translation) is involved in the media path, since endpoints can not learn their media socket's NATed (server-reflexive) addresses. There are various techniques to solve this problem:

- **STUN (Simple Traversal of UDP over NATs) protocol** - helps determine server reflexive address (works only with full cone NAT)
- **TURN (Traversal Using Relay NAT) protocol** - relays data through a public server (works for all NAT types)

The optimal communication path between endpoints in all possible NAT/firewall scenarios can be assured with the combination of the above methods. ICE collects all possible endpoint addresses for a communication channel to form "candidates" (local addresses, server reflexive addresses, relay address, etc.) at both endpoint, it assigns priority to each candidate, changes candidate list between endpoints, and starts a connectivity check by pinging each other via all possible combinations of the candidates. The highest priority successful connection is used during the data exchange. Local addresses have the highest priority (forces direct p2p connection if possible), and relayed addresses the lowest (if others fail the relayed communication is chosen because of the extra bandwidth, delay and jitter costs at the relay host).

V. REFERENCES

[1] Alex Lewis, Andrew Abbate and Tom Pacyk, Microsoft Lync Server 2010 Unleashed, Pearson Education, Inc. 2011

[2] Nathan Winters and Keith Hanna, Mastering Microsoft Lync Server 2010, John Wiley & Sons, Inc. 2012

[3] Peter Krasznai, Edge Server Based Call Recording for Microsoft Lync, 2012

[4] Image sources: Verba Technologies, LLC. and Microsoft Lync client