

# eXpress

Communication  
System

## Administrator's Guide

### Volume 1. Installation

Build 3.36  
03/17/2025



© Unlimited Production, 2025. All rights reserved.

All copyrights to the operating documentation are protected.

This document is included in the product delivery package. It is subject to all terms and conditions of the license agreement. Neither this document, nor any part thereof, whether printed or electronic, may be copied or transmitted to third parties for commercial purposes without the express written permission of Unlimited Production.

The information contained in this document may be changed by the developer without special notice, which does not constitute a breach of obligations to the user by Unlimited Productions.

The server addresses, configuration file values, and user account data specified in the document are provided for example and are for informational purposes only. User data, including biometric data, are fictitious and do not contain personal data.

The provided components of eXpress CS as part of the delivery are intended exclusively for demonstration of functionality and are not intended for operation in a productive environment. For the correct functioning of eXpress CS, it is necessary to develop an architectural scheme of the installation taking into account the specifics of the infrastructure for productive operation.

Mailing address:	127030, Moscow, 24/1 Novoslobodskaya Street
Phone:	+7 (499) 288-01-22
E-mail:	<a href="mailto:sales@express.ms">sales@express.ms</a>
Web:	<a href="https://express.ms/">https://express.ms/</a>

## TABLE OF CONTENTS

<b>INTRODUCTION .....</b>	<b>6</b>
<b>TERMS AND DEFINITIONS .....</b>	<b>7</b>
<b>CHAPTER 1 .....</b>	<b>8</b>
<b>GENERAL INFORMATION .....</b>	<b>8</b>
<b>Purpose of the System.....</b>	<b>8</b>
<b>Main Functions .....</b>	<b>8</b>
<b>Main Components .....</b>	<b>8</b>
<b>Available Roles .....</b>	<b>10</b>
<b>Architecture.....</b>	<b>12</b>
Regional server .....	12
Single Corporate Server.....	14
Decoupled Corporate Server .....	17
Enterprise Server and Single Corporate Server.....	20
Enterprise Server and Decoupled Enterprise Server .....	22
<b>Types of authentication .....</b>	<b>24</b>
Authentication with Active Directory .....	24
Authentication with ADLDS.....	25
Authentication with e-mail.....	26
Authentication with Keycloak .....	27
<b>System requirements .....</b>	<b>28</b>
Platform requirements .....	28
DNS Requirements .....	34
Certificate requirements.....	34
LDAP Corporate Directory Requirements.....	35
SMTP server requirements .....	35
Media Server Requirements.....	35
Network communication requirements .....	35
Web Client Server Requirements .....	36
Requirements for Storing Videoconferencing Recording Files .....	36
DLPS Requirements.....	36
<b>CHAPTER 2 .....</b>	<b>38</b>
<b>INSTALLATION .....</b>	<b>38</b>
<b>Pre-Configuration .....</b>	<b>38</b>
Ubuntu/Debian OS .....	39
Centos/RHEL Operating System .....	41
Astra Linux Eagle Operating System .....	42
<b>Installing ETS .....</b>	<b>44</b>
<b>Web Client Installation .....</b>	<b>46</b>
<b>Media Server Installation .....</b>	<b>47</b>
Pre-Configuration.....	48

Media Server Installation .....	48
<b>Transcoding Server Installation.....</b>	<b>50</b>
<b>Corporate Server Installation .....</b>	<b>53</b>
Single CTS Installation.....	53
Front CTS and Back CTS Installation .....	55
<b>Connecting Media Server to CTS .....</b>	<b>59</b>
<b>Setting Up Media Server .....</b>	<b>59</b>
Setting Up JANS, STUN and TURN Servers.....	59
Setting up IP telephony .....	61
<b>Links Server Installation .....</b>	<b>64</b>
<b>Installing DLPS.....</b>	<b>65</b>
Installing DLPS on a Dedicated Server .....	65
Installing DLPS on Single CTS.....	67
Installing DLPS on Single CTS with keys stored on external media .....	68
<b>Installing Call and Conference Recording Components.....</b>	<b>69</b>
<b>Certificate Verification.....</b>	<b>70</b>
<b>Starting up the server.....</b>	<b>70</b>
<b>CHAPTER 3 .....</b>	<b>72</b>
<b>SETTING UP THE SERVER.....</b>	<b>72</b>
<b>SETTING UP THE ETS SERVER.....</b>	<b>72</b>
Connecting the TLS Certificate .....	73
Setting Up Video and Voice Communication .....	73
Connecting the SMTP Server.....	73
Setting Up Push Notifications .....	74
Setting Up SMS Service .....	80
Setting Up Administrator Authentication.....	83
Setting Up Connections for Corporate Servers.....	84
<b>Setting Up CTS.....</b>	<b>86</b>
Connecting the TLS Certificate and the Botx SSL Certificate .....	86
Setting Up Video and Voice Communication .....	88
Connecting the SMTP Server.....	88
Setting Up Administrator Authentication.....	89
Setting Up Registration .....	91
Setting Up Trusted Connections .....	99
<b>CHAPTER 4 .....</b>	<b>102</b>
<b>UPDATE PROCEDURE .....</b>	<b>102</b>
<b>CHAPTER 5 .....</b>	<b>103</b>
<b>TROUBLESHOOTING TYPICAL ERRORS .....</b>	<b>103</b>
<b>CHAPTER 6 .....</b>	<b>106</b>
<b>ELIMINATING VULNERABILITIES .....</b>	<b>106</b>
<b>APPENDIX 1.....</b>	<b>108</b>
<b>SINGLE CTS NETWORK INTERACTIONS .....</b>	<b>108</b>

<b>APPENDIX 2.....</b>	<b>110</b>
<b>FRONT CTS, MEDIA AND BACK CTS NETWORK INTERACTIONS .....</b>	<b>110</b>
<b>APPENDIX 3.....</b>	<b>113</b>
<b>ETS, MEDIA AND SINGLE CTS NETWORK INTERACTIONS .....</b>	<b>113</b>
<b>APPENDIX 4.....</b>	<b>115</b>
<b>ETS, MEDIA, FRONT CTS AND BACK CTS SERVER NETWORK INTERACTIONS ....</b>	<b>115</b>
<b>APPENDIX 5.....</b>	<b>118</b>
<b>MONITORING OF EXPRESS OPERATION.....</b>	<b>118</b>
Prometheus .....	118
Grafana .....	122
Alerts .....	124
<b>APPENDIX 6.....</b>	<b>128</b>
<b>SETTING UP SMARTAPPPROXY HOSTS.....</b>	<b>128</b>
<b>APPENDIX 7.....</b>	<b>130</b>
<b>DIAGRAM OF SINGLE CTS NETWORK INTERACTIONS .....</b>	<b>130</b>
<b>APPENDIX 8.....</b>	<b>131</b>
<b>ATE NETWORKING DIAGRAM FOR THE DEPLOYMENT OF THE FRONT CTS AND BACK CTS SERVERS .....</b>	<b>131</b>
<b>APPENDIX 9.....</b>	<b>132</b>
<b>CTS AND KEYCLOAK INTEGRATION .....</b>	<b>132</b>
<b>Keycloak Requirements .....</b>	<b>132</b>
<b>Registration/Authorization Steps .....</b>	<b>133</b>
<b>Network Interactions .....</b>	<b>133</b>
<b>Setting Up Integration.....</b>	<b>134</b>
Creating Client Scope .....	135
Setting Up Field Mapping .....	136
Creating Client .....	139
Setting Up Keycloak Authorization Form Display .....	140
Setting Up QR Code Authorization .....	141
Role Model .....	141
<b>CHANGE HISTORY.....</b>	<b>146</b>

## INTRODUCTION

This manual is intended for administrators of the product eXpress Communication System (hereinafter referred to as eXpress CS, eXpress, system). This Volume 1 of the Administrator's Guide contains information necessary for the installation and configuration of the system.

**Product Support Service** You can contact the product support service by e-mail [support@express.ms](mailto:support@express.ms). The page of the product support service on the Unlimited Production website is available at <https://express.ms/faq/>.

**Website.** Information on the product by Unlimited Production can be found on the website <https://express.ms/>.

List of volumes of the Administrator's Guide:

- Volume 1. "Administrator's Guide. Installation.
- Volume 2. "Administrator's Guide. Operation of the CTS Server.
- Volume 3. "Administrator's Guide. Operation of the ETS Server.
- Volume 4. "Administrator's Guide. Installation and Operation of the RTS Server (available upon request).

## TERMS AND DEFINITIONS

Term	Definition
AD	Active Directory is Microsoft Corporation's directory service for the Windows Server operating systems
API	Application programming interface — the interface enabling communication between software programs and applications
APNS	Apple Push Notification Service
botX	A platform for chatbot development
CTS	Corporate Transport Server
ETS	Enterprise Transport Server
FCM	Firebase Cloud Messaging is a service that simplifies messaging between mobile apps and server apps
JSON	JavaScript-based text-based data interchange format
NTLM	Network Authentication Protocol developed by Microsoft for Windows NT
RTS	Regional Transport Server
SIEM	Security information and event management
Single CTS	Single Corporate Server
SIP	Session Initiation Protocol — a data transfer protocol, describing the method of establishing and terminating a user Internet session involving the exchange of multimedia content(IP telephony, video and audio conferencing, and instant messaging)
SmartApp	SmartApp is a web application, which is implemented as an add-on, executed inside the application, and designed for accessing corporate services and systems.
SMTP	A network protocol designed for e-mail transmission in TCP/IP networks
SSL	Cryptographic protocol for secure communication
STUN	A network protocol for discovering an external IP address, used to establish a UDP connection between two hosts when they are both behind a NAT router.
TLS	Transport Layer Security Protocol
TTS	Transport Transfer Server. A server designed to transmit messages between corporate servers instead of the RTS server, including between the CTS servers that do not have a trusted connection with each other (non-trusted CTS servers)
TURN	A protocol for receiving incoming data over TCP or UDP connections
VAPID keys	Voluntary Application Server Identification involves a pair of keys: a public one and a private one. The private key is kept secret by the server whereas the public key is passed to the client. The keys allow the push notification service to know which application server signed the user and to be sure that it is the same server that sends notifications to a particular user.
ATE	A company's automatic telephone exchange
Widget	A structural element of the panel responsible for visual display of a part of information collected by the system.
Videoconferencing	Multicast videoconferencing
Cache	A fast-access intermediate buffer containing information that is most likely to be queried
CTN	Corporate data transmission network
TSP	Trusted Services Platform
PC	Personal Computer
Decoupled CTS	Decoupled corporate server Front CTS and Back CTS
Routing	The contour, in which a chat exists (corporate, public, mixed)
Trust	A service for data transfer between the CTS server and the RTS server and other services within their contour

# Chapter 1

## GENERAL INFORMATION

### PURPOSE OF THE SYSTEM

eXpress CS is designed to provide high-quality continuous communication between the company's employees and to reduce the risk of information leaks by moving the exchange channels from the Internet into the perimeter of the Company's local computer networks.

### MAIN FUNCTIONS

eXpress CS performs the following main functions:

- enabling fast exchange of text messages and files by users with the help of mobile devices and the web client on PCs within personal and group chats;
- ensuring secure storage and transmission of confidential data;
- creating copies of data to restore the subsystem's functionality when it is damaged or destroyed;
- streamlining the use of resources;
- making personal and group audio and video calls;
- recording of calls and video conferences.

### MAIN COMPONENTS

eXpress CS envisages three user interaction contours (which can be supplied in three versions):

- public (external);
- enterprise contour (company's internal contour, which combines several internal servers);
- corporate (internal).

The public (external) user interaction contour is used for the following purposes:

- initial user registration;
- sending push notifications;
- exchanging messages and files with users who are not connected to any internal contour;
- making calls by users not connected to any internal contour;
- routing messages and files between internal contours that have no direct trusted connections.

The enterprise contour (company's internal contour) is used for:

- user registration;
- sending push notifications;
- routing messages and files between corporate contours that have no direct trusted connections.



The corporate (internal) user interaction contour is used for the following purposes:

- registering corporate users;
- exchanging messages, files and making calls to corporate users;
- providing a corporate address book;
- routing messages and files between the company's corporate contour and corporate contours of partners, with whom trusted connections are established.

eXpress CS incorporates the following separately installed components:

- regional eXpress server (hereinafter referred to as the "RTS server");
- enterprise server (hereinafter referred to as the "ETS server");
- corporate eXpress server (hereinafter referred to as the "CTS server");
- Media server;
- Bot Server;
- Mobile app;
- Desktop app;
- Web app.

---

**Attention!** For all the described functions to work properly, the application and server versions must match.

---

The RTS, ETS and CTS servers are the main elements in the system architecture.

The RTS brings together and maintains computer networks within one region and is responsible for the operation of the public interaction contour.

The ETS server brings together and maintains computer networks and corporate servers within one large company and is responsible for the operation of the enterprise contour.

A customized application is released for the ETS server, which is managed by the company operating the ETS. The CTS server users, which are connected to the ETS server, receive SMS messages and push notifications from this ETS server (for more details, see the document "Administrator's Guide. Volume 3. Operation of the ETS Server").

The CTS server connects and maintains client devices within the organization, connects to the ETS server or the RTS server and acts as an intermediary between the client device and the ETS/RTS server. The CTS server is responsible for the operation of the enterprise contour. With the ETS sever installed, information exchange between corporate servers takes place within the enterprise; data from the CTS server is transmitted to the ETS server, the ETS server performs information exchange with the external contour (for more details see the document "Administrator's Guide. Volume 2. Operation of the CTS Server").

The client device can connect to both the CTS server and the ETS server or the RTS server directly. For each server, a user registers their profile. Depending on the active profile, the user has access to their resources in the form of chats, contacts and messaging history. The client can connect to the CTS server once a connection to the RTS or ETS server has been established. All messages transferred between corporate users are stored on the CTS server in encrypted form and are not accessible to server administrators.

A separate Media server is used to support voice and video calls.

If the number of users is 100 or more, the Transcoding server is detached from the Media server to a separate server.

For the deployment of chatbots and SmartApps, a separate server (Bot Server) is used.

To integrate the ATE system, the SIP-telephony module is used, which allows making and receiving voice calls, maintaining the phone book and matching users with ATE numbers ("Caller ID").

Comparison of system functions and features is described in [Table 1](#):

*Table 1*

Functions	Features
Outgoing call	<ul style="list-style-type: none"> <li>• Making voice calls to the ATE using a mobile device or PC;</li> <li>• Calling a subscriber by dialing a number</li> </ul>
Incoming call	Receiving voice calls from the ATE using a mobile device or PC
Maintaining a phone book	Integration of the telephone book of the telephony module: <ul style="list-style-type: none"> <li>• the phone book of the device on which eXpress CS is installed;</li> <li>• entries saved in eXpress CS;</li> <li>• AD entries</li> </ul>
Caller ID	Matching the number of a calling subscriber with a corresponding user of eXpress CS when an incoming call is received from the ATE to the device with eXpress CS installed. As a result, the called user receives information about the caller (name, avatar, etc.). When making an outgoing call from a device with installed eXpress CS to the ATE the called user is automatically detected and information about them is displayed

For integration with data leak prevention systems that check user messages for prohibited content, the ICAP protocol (TCP/1344 port) is used.

The system is managed via administrator web interface, which makes it possible to set up eXpress and control the operation of the application.

## AVAILABLE ROLES

The system is managed by the organization's employees with administrator rights. The administrative rights in the system are assigned hierarchically.

For the safe and successful operation of eXpress, the following roles are identified (see [Table 2](#)):

*Table 2*

Role	Rights	Account type
Administrator	<ul style="list-style-type: none"> <li>• role assignment;</li> <li>• viewing security log;</li> <li>• managing chats;</li> <li>• managing user accounts;</li> <li>• connecting chatbots;</li> <li>• managing system settings</li> </ul>	Internal user
Corporate User	<ul style="list-style-type: none"> <li>• sending messages;</li> <li>• creating a chat;</li> <li>• viewing the server address book;</li> <li>• connecting to chatbots</li> </ul>	Internal user
Regional User	<ul style="list-style-type: none"> <li>• sending messages;</li> <li>• creating a chat</li> </ul>	External user

Role	Rights	Account type
Security Administrator <sup>1</sup>	<ul style="list-style-type: none"> <li>viewing messages in the DLP console;</li> <li>viewing logs in the DLP console</li> </ul>	Internal user

The type of account depends on the position of the server on which the user is authorized. If there is a RTS server within the protected contour, a regional user becomes an internal user.

eXpress CS envisages the creation of administrators with limited rights for specific tasks.

Administrators' tasks:

- installation and management of updates of system-wide and application software;
- configuration, maintenance and monitoring of server equipment;
- backup management and data recovery;
- centralized configuration of the Mobile app;
- managing user accounts.

On the CTS server, within the framework of the role model for individual user groups, the administrator can set restrictions for users with regard to operations with attachments:

- prohibition of sending/forwarding attachments to chats;
- prohibition of downloading/viewing attachments in chats;
- prohibition of the ability to forward/share/save attachments to the device's memory.

First, the administrator creates user groups in the User Groups section to which the restrictions will apply, and then, in the Role Model section, sets the rules that the restrictions will be subject to.

Restrictions can be configured for specific users or specific groups based on server affiliation (for more information, see the document "Administrator's Guide. Volume 2. Operation of the CTS Server").

---

<sup>1</sup> for CTS server users only (for more details, see the document "Administrator's Guide. Volume 2. Operation of the CTS Server").

## ARCHITECTURE

**Note.** This document covers a non-fault-tolerant product configuration. For information about fault-tolerant configuration options, please contact the developer.

eXpress CS consists of the external contour and the internal contour. The external and internal contours of the product are connected in the local network by means of a special service – trust. The external contour consists of the regional server (RTS), the internal contour consists of the corporate server (CTS) or enterprise server (ETS) and the CTS servers that connect to it.

The server part of eXpress is based on a microservice architecture using Docker-based containerization. This solution maximizes the automation of the deployment and upgrade of the eXpress server software.

The CTS server supports the following two types of deployment:

- [Single Corporate Server \(Single CTS\)](#);
- [Decoupled Corporate Server \(Front CTS and Back CTS\)](#).

The ETS server supports the following two types of deployment:

- [Single ETS and single eXpress Server \(Single CTS\)](#);
- [Single ETS and decoupled eXpress Server \(Front CTS and Back CTS\)](#).

The Audio Exchange Server (Media server) is hosted on the Internet or in the company's perimeter network.

If the number of users is 100 or more, the Transcoding server is detached from the Media server to a separate server. The Transcoding server is located on the company's internal network.

The chatbot server (Bot Server) is hosted on the company's internal network and is designed to host chatbots and the necessary components for their operation, such as databases. Connection to the Bot server is performed using the botx docker container.

## REGIONAL SERVER

**Important!** Installation and configuration of the RTS server shall be performed exclusively by employees of the developer company. The information about the RTS server provided in this document is for informational purposes only.

For all system deployment options, the Regional Server (RTS) is hosted on the Internet and includes the following containers:

- admin (administrator interface);
- audit (connection audit service);
- authentication\_service (responsible for authorization on RTS);
- conference\_bot (a bot for notifications about upcoming conferences; sends a link to a saved recording when making personal calls);
- email\_notifications (responsible for sending e-mail messages with the authentication code);
- etcd (add-on to settings, responsible for storing service settings);
- events (service for informing users about events in chats);
- file\_service (file upload service);
- kafka (message handler between services);

- kafka\_exporter (responsible for extracting metrics from Kafka);
- kdc (key storage);
- messaging (messaging service responsible for connecting clients via the websocket protocol);
- nginx (web server that accepts connections from the outside and is responsible for routing internal connections);
- notifications\_bot (a bot for sending messages to the global chat);
- phonebook (address directory);
- postgres (the main services database);
- postgres\_exporter (responsible for extracting metrics from postgres);
- preview\_service (service for previewing pages to which links are sent);
- prometheus (responsible for removing, processing and storing service metrics);
- push\_service (service for sending push notifications);
- redis (KV storage);
- redis\_exporter (responsible for removing metrics from redis);
- routing\_schema\_service (service for building routing diagrams, visualises the routing diagram in chats);
- settings (responsible for storing service settings);
- sms\_service (service for sending text messages (SMS));
- stickers (service for managing stickers);
- trusts (responsible for interaction with the ETS and CTS servers);
- voex (service for making audio calls);
- docker\_socket\_proxy (responsible for viewing container logs in the administrator interface);
- traefik (responsible for receiving all external connections);
- botx (responsible for bot integration);
- metrics\_service (service for collecting individual indicators).

Media project composition:

- coturn (STUN/TURN server);
- janus (group call service).

## SINGLE CORPORATE SERVER

A typical Single CTS, Media and Transcoding server deployment diagram is depicted below (see [Figure 1](#)).

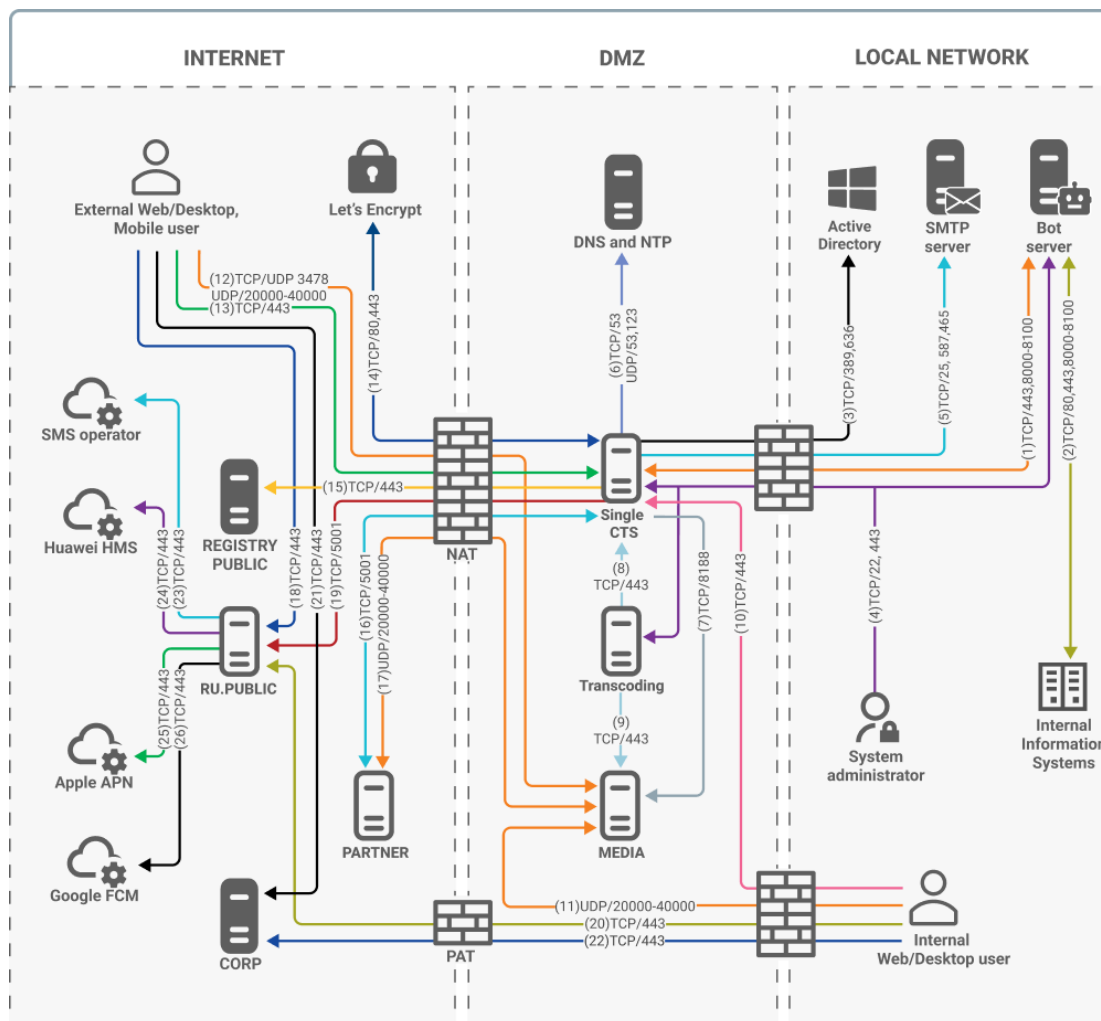


Figure 1. A typical Single CTS and Media server deployment diagram

**Attention!** Partner is the partner CTS server with which a trusted connection can be established. It is located in the local network of another organization or on the Internet. The users of such a server participate in audio and video calls with users of the CTS server, therefore relevant ports must be opened for media exchange via the SRTP protocol.

The numbers of network interactions correspond to the number of the line in [Appendix 1](#).

ATE network interactions diagram for Single CTS deployment scheme and network interactions for this deployment scheme are provided in [Appendix 8](#).

Single CTS consists of two different projects: Media and CTS.

Single CTS is hosted in the company's perimeter network and includes the following Docker containers:

- ad\_integration (integrates with Active Directory and other LDAP services, is responsible for client authorization using NTLM and AD);
- admin (administrator interface);

- apigw (service for informing users about events in chats);
- audit (connection audit service);
- botx (responsible for bot integration);
- conference\_bot (a bot for notifications about upcoming conferences; sends a link to a saved recording when making personal calls);
- corporate\_directory (catalog of open bots and chats);
- docker\_socket\_proxy (responsible for restricting access to the Docker socket);
- email\_notifications (responsible for sending e-mail messages with the authentication code);
- etcd (add-on to settings, responsible for storing service settings);
- events (service for informing users about events in chats);
- file\_service (file upload service);
- kafka (message handler between services);
- kafka\_exporter (responsible for extracting metrics from Kafka);
- kdc (key storage);
- messaging (messaging service responsible for connecting clients via the websocket protocol);
- metrics\_service (service for collecting individual indicators of the ETS/CTS servers);
- nginx (web server that accepts connections from the outside and is responsible for routing internal connections);
- notifications\_bot (a bot for sending messages to the global chat);
- phonebook (address directory);
- postgres (the main services database);
- postgres\_exporter (responsible for extracting metrics from postgres);
- prometheus (responsible for removing, processing and storing service metrics);
- redis (KV storage)<sup>1</sup>;
- redis\_exporter (responsible for removing metrics from redis);
- routing\_schema (service for building routing diagrams, visualises the routing diagram in chats);
- settings (responsible for storing service settings);
- smartapp\_proxy (responsible for exchanging files between SmartApp and the CTS server);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input);
- transcoding\_manager (manages the encoding process);

---

<sup>1</sup>It is recommended to use a separate system Redis during installation. The embedded Redis container is intended for demonstrations of the product's capabilities.

- trusts (responsible for the exchange of events between the RTS, ETS and CTS servers);
- preview\_service (service for previewing pages to which links are sent);
- homescreen (SmartApp, which provides the user with access to a single virtual space, which contains corporate services, a news feed and announcements of upcoming events);
- stickers (service for managing stickers);
- roles (role model);
- recordings\_bot (a bot that sends a link to the recording file once the encoding process is complete);
- voex (service for making audio calls).

Media server is hosted in the company's perimeter network and includes the following Docker containers:

- coturn (STUN/TURN service);
- janus (group call service).

The Transcoding server is hosted in the company's demilitarized network zone and contains a transcoding container (which is responsible for recoding the recording into the output format).

The DLPS service is supplied separately and consists of the following containers:

- dlps (DLP system);
- nginx (web server that accepts connections from the outside and is responsible for routing internal connections);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input);
- prometheus (responsible for removing, processing and storing service metrics).

The Links service is supplied separately and consists of the following containers:

- link (responsible for redirecting the user to a chat, channel, conference, call);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input).



## DECOUPLED CORPORATE SERVER

A typical Front CTS, Media, Transcoding, and Back CTS server deployment diagram is depicted below (see [Figure 2](#)).

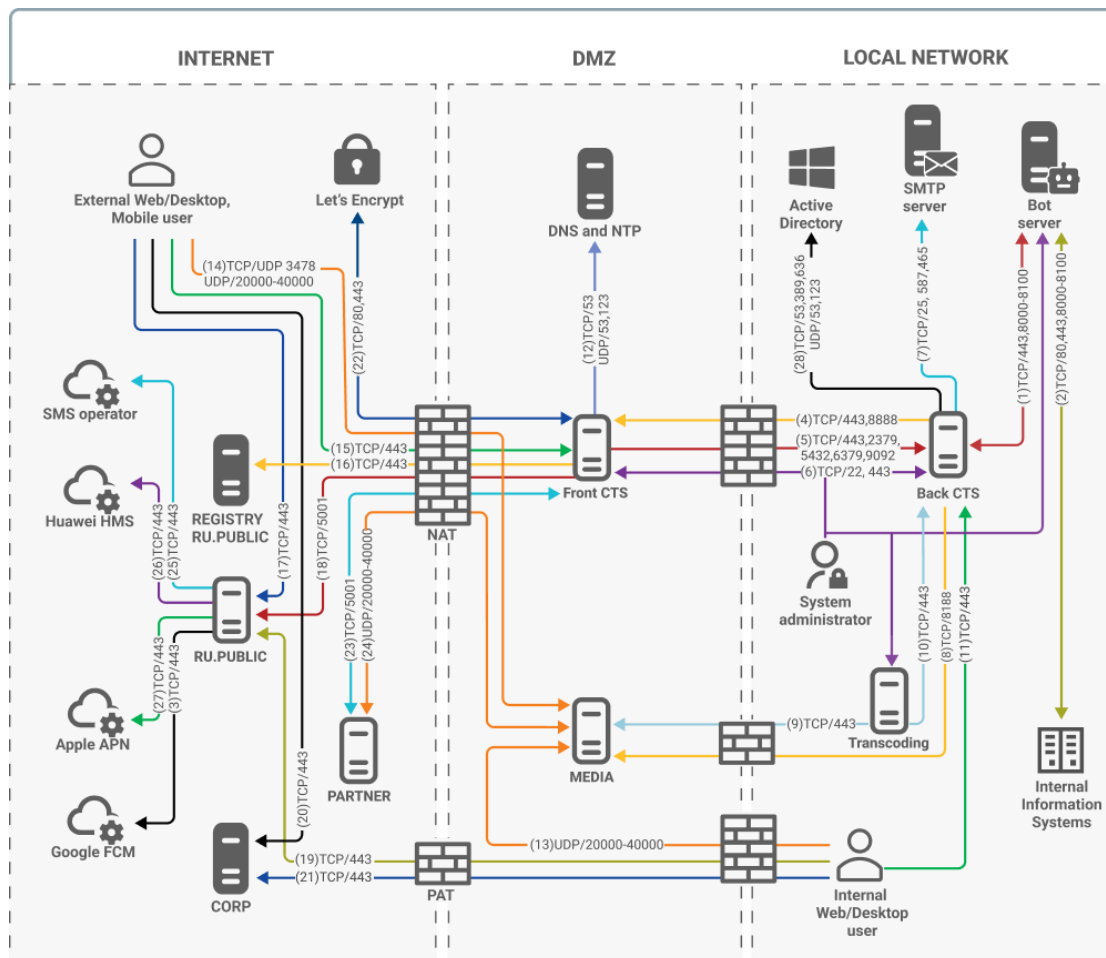


Figure 2. A typical decoupled CTS (Front Media/Back server) deployment diagram

**Attention!** Partner is the partner CTS server with which a trusted connection can be established. It is located in the local network of another organization or on the Internet. The users of such a server participate in audio and video calls with users of the CTS server, therefore relevant ports must be opened for media exchange via the SRTP protocol.

The numbers of network interactions correspond to the number of the line in [Appendix 2](#).

The decoupled server consists of the Front CTS and Back CTS servers.

ATE network interactions diagram with Front CTS + Media and Back CTS servers and network interactions for this deployment scheme are provided in [Appendix 9](#).

Front CTS consists of two different projects: Media and CTS.

The Front CTS server is hosted in the company's perimeter network and includes the following Docker containers:

- nginx (web server that accepts connections from the outside and is responsible for routing internal connections);
- prometheus (responsible for removing, processing and storing service metrics);

- traefik (responsible for receiving certificates from LE and terminating TLS at the input);
- trusts (ensures interaction with the ETS/RTS server and other trusted corporate CTS servers).

The Back CTS server is hosted on the company's local network and includes the following docker containers:

- ad\_integration (integrates with Active Directory and other LDAP services, is responsible for client authorization using NTLM and AD);
- admin (administrator interface);
- audit (connection audit service);
- apigw (service for informing users about events in chats);
- botx (responsible for bot integration);
- conference\_bot (a bot for notifications about upcoming conferences; sends a link to a saved recording when making personal calls);
- corporate\_directory (catalog of open bots and chats);
- docker\_socket\_proxy (responsible for restricting access to the Docker socket);
- email\_notifications (responsible for sending e-mail messages with the authentication code);
- etcd (add-on to settings, responsible for storing service settings);
- events (service for informing users about events in chats);
- file\_service (file upload service);
- kafka (message handler between services);
- kafka\_exporter (responsible for extracting metrics from Kafka);
- kdc (key storage);
- messaging (messaging service responsible for connecting clients via the websocket protocol);
- metrics\_service (service for collecting individual indicators of the ETS/CTS servers);
- nginx (web server, Connect for internal routing of connections);
- notifications\_bot (a bot for sending messages to the global chat);
- phonebook (address directory);
- postgres (the main services database);
- postgres\_exporter (responsible for extracting metrics from postgres);
- prometheus (responsible for removing, processing and storing service metrics);
- recordings\_bot (a bot that sends a link to the recording file once the encoding process is complete);
- redis (KV storage)<sup>1</sup>;

---

<sup>1</sup>It is recommended to use a separate system Redis during installation. The embedded Redis container is intended for demonstrations of the product's capabilities.

- redis\_exporter (responsible for removing metrics from redis);
- routing\_schema (service for building routing diagrams, visualises the routing diagram in chats);
- smartapp\_proxy (responsible for exchanging files between SmartApp and the CTS server);
- homescreen (SmartApp, which provides the user with access to a single virtual space, which contains corporate services, a news feed and announcements of upcoming events);
- settings (responsible for storing service settings);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input);
- transcoding\_manager (manages the encoding process);
- preview\_service (service for previewing pages to which links are sent);
- stickers (service for managing stickers);
- roles (role model);
- voex (service for making audio calls).

Media server is hosted in the company's perimeter network and includes the following Docker containers:

- coturn (STUN/TURN service);
- janus (group call service).

The Transcoding server is hosted in the company's local network and contains a transcoding container (which is responsible for recoding the recording into the output format).

The DLPS service is supplied separately and consists of the following containers:

- dlps (DLP system);
- nginx (web server that accepts connections from the outside and is responsible for routing internal connections);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input);
- prometheus (responsible for removing, processing and storing service metrics).

The Links service is supplied separately and consists of the following containers:

- link (responsible for redirecting the user to a chat, channel, conference, call);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input).

## ENTERPRISE SERVER AND SINGLE CORPORATE SERVER

A typical ETS, Single CTS, Media, Transcoding, and Web Client server deployment diagram is depicted below (see [Figure 3](#)).

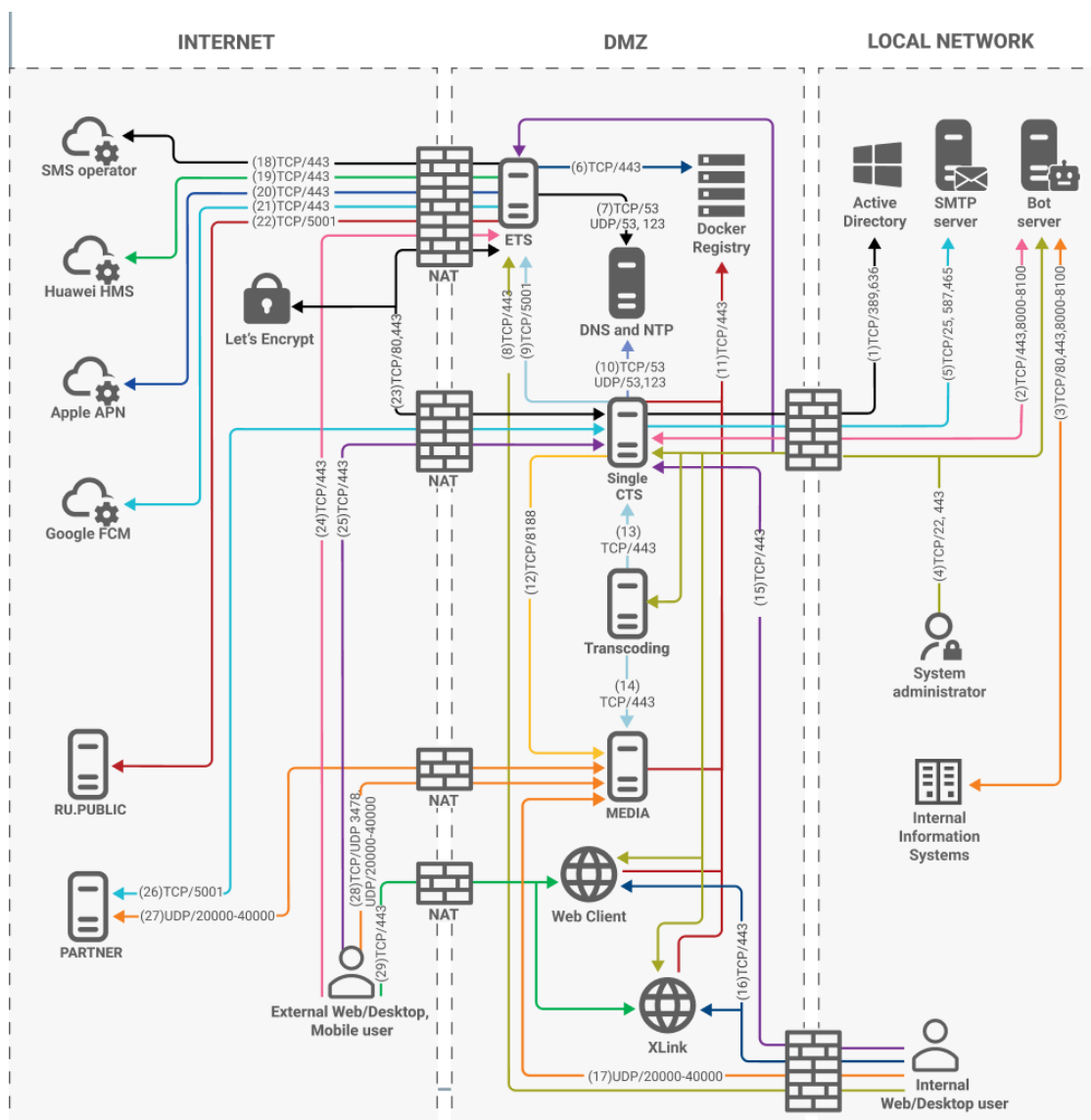


Figure 3. A typical ETS, Single CTS, Media and Web Client server deployment diagram

**Attention!** Partner is the partner CTS server with which a trusted connection can be established. It is located in the local network of another organization or on the Internet. The users of such a server participate in audio and video calls with users of the CTS server, therefore relevant ports must be opened for media exchange via the SRTP protocol.

The numbers of network interactions correspond to the number of the line in [Appendix 3](#).

The ETS is hosted in the company's perimeter network and includes the following docker containers:

- audit (connection audit service);
- admin (administrator interface);
- logstack (centralized log processing);

- authentication\_service (responsible for authorization on the ETS and RTS servers);
- botx (responsible for bot integration);
- email\_notifications (responsible for sending messages with the authentication code via e-mail);
- etcd (add-on to settings, responsible for storing service settings);
- events (service for informing users about events in chats);
- file\_service (file upload service);
- janus (group call service);
- kafka (message handler between services);
- kafka\_exporter (responsible for extracting metrics from Kafka);
- kdc (key storage);
- messaging (messaging service responsible for connecting clients via the websocket protocol);
- metrics\_service (service for collecting individual indicators of the ETS/CTS servers);
- nginx (web server that accepts connections from the outside and is responsible for routing internal connections);
- notifications\_bot (a bot for sending messages to the global chat);
- conference\_bot (a bot for notifications about upcoming conferences, sends a link to a saved recording when making personal calls);
- phonebook (address directory);
- postgres (the main services database);
- postgres\_exporter (responsible for extracting metrics from postgres);
- preview\_service (service for previewing pages to which links are sent);
- prometheus (responsible for removing, processing and storing service metrics);
- push\_service (service for sending push notifications);
- redis (KV storage)<sup>1</sup>;
- redis\_exporter (removing metrics from redis);
- settings (responsible for storing service settings);
- sms\_service (service for sending text messages (SMS));
- stickers (service for managing stickers);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input);
- trusts (responsible for interaction with the RTS and CTS servers);
- docker\_socket\_proxy (responsible for viewing container logs in the administrator interface);
- voex (service for making audio calls).

---

<sup>1</sup>It is recommended to use a separate system Redis during installation. The embedded Redis container is intended for demonstrations of the product's capabilities.

The Web Client server is hosted in the company's perimeter network and includes the following docker containers:

- web\_client (web client service);
- link (a service that provides links to conferences).

The list of Single CTS, Media, and Transcoding server containers is provided in the subsection [Single Corporate Server](#).

## ENTERPRISE SERVER AND DECOUPLED ENTERPRISE SERVER

A typical ETS, Front CTS, Back CTS, Media, Transcoding, and Web Client server deployment diagram is depicted below (see [Figure 4](#)).

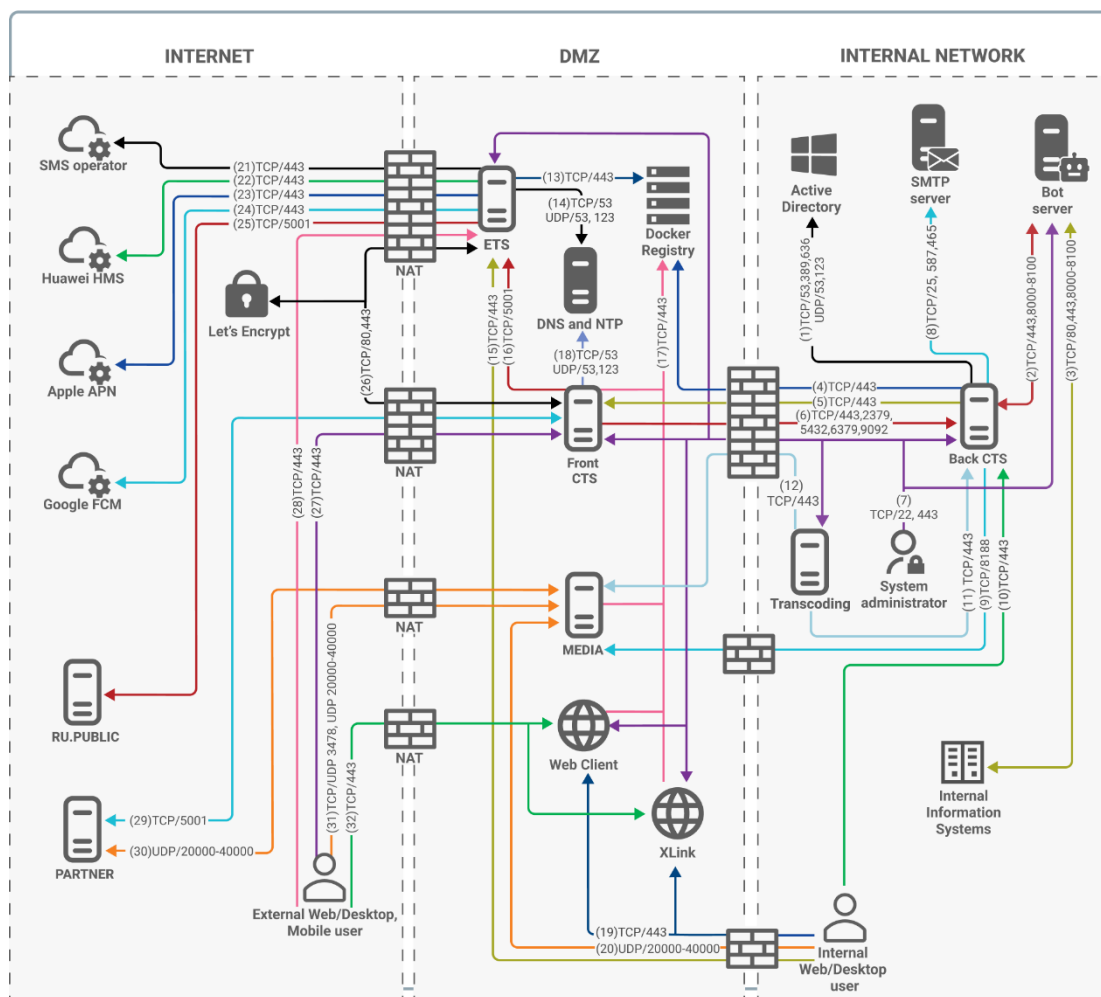


Figure 4. A typical ETS, Front CTS, Back CTS, Media and Web Client server deployment diagram

**Attention!** Partner is the partner CTS server with which a trusted connection can be established. It is located in the local network of another organization or on the Internet. The users of such a server participate in audio and video calls with users of the CTS server, therefore relevant ports must be opened for media exchange via the SRTP protocol.

The numbers of network interactions correspond to the number of the line in [Appendix 4](#).

The ETS is hosted in the company's perimeter network and includes the following docker containers:

- audit (connection audit service);

- admin (administrator interface);
- logstack (centralized log processing);
- authentication\_service (responsible for authorization on the ETS and RTS servers);
- botx (responsible for bot integration);
- email\_notifications (responsible for sending messages with the authentication code via e-mail);
- etcd (add-on to settings, responsible for storing service settings);
- events (service for informing users about events in chats);
- file\_service (file upload service);
- janus (group call service);
- kafka (message handler between services);
- kafka\_exporter (responsible for extracting metrics from Kafka);
- kdc (key storage);
- messaging (messaging service responsible for connecting clients via the websocket protocol);
- metrics\_service (service for collecting individual indicators of the ETS/CTS servers);
- nginx (web server that accepts connections from the outside and is responsible for routing internal connections);
- notifications\_bot (a bot for sending messages to the global chat);
- conference\_bot (a bot for notifications about upcoming conferences; sends a link to a saved recording when making personal calls);
- phonebook (address directory);
- postgres (the main services database);
- postgres\_exporter (responsible for extracting metrics from postgres);
- preview\_service (service for previewing pages to which links are sent);
- prometheus (responsible for removing, processing and storing service metrics);
- push\_service (service for sending push notifications);
- redis (KV storage)<sup>1</sup>;
- redis\_exporter (removing metrics from redis);
- settings (responsible for storing service settings);
- sms\_service (service for sending text messages (SMS));
- stickers (service for managing stickers);
- traefik (responsible for receiving certificates from LE and terminating TLS at the input);
- trusts (responsible for interaction with the RTS and CTS servers);

---

<sup>1</sup>It is recommended to use a separate system Redis during installation. The embedded Redis container is intended for demonstrations of the product's capabilities.

- `docker_socket_proxy` (responsible for viewing container logs in the administrator interface);
- `voex` (service for making audio calls).

The Web Client server is hosted in the company's perimeter network and includes the following docker containers:

- `web_client` (web client service);
- `link` (a service that provides links to conferences).

The list of detached CTS, Media, and Transcoding server containers is provided in the subsection [Decoupled Corporate Server](#).

## TYPES OF AUTHENTICATION

eXpress CS supports several types of authentication:

- [with Active Directory](#);
- [with ADLDS](#);
- [with e-mail](#);
- [with Keycloak](#).

---

### AUTHENTICATION WITH ACTIVE DIRECTORY

An Active Directory domain controller is connected to the CTS server directly (or through a VPN tunnel). User authentication is performed with a login/password pair from Active Directory

Configuration with the export of users from AD is supported. In this configuration, authentication is performed using the PIN code sent to your email. This configuration allows quick management of the CTS server account behavior by modifying the account in Active Directory. For example: disconnecting, expiring, changing the password, and excluding the account from the sample group in Active Directory.

A connection example is provided based on a typical corporate server deployment scheme ([Figure 5](#)).

The numbers of network interactions correspond to the number of the line in [Appendix 2](#).



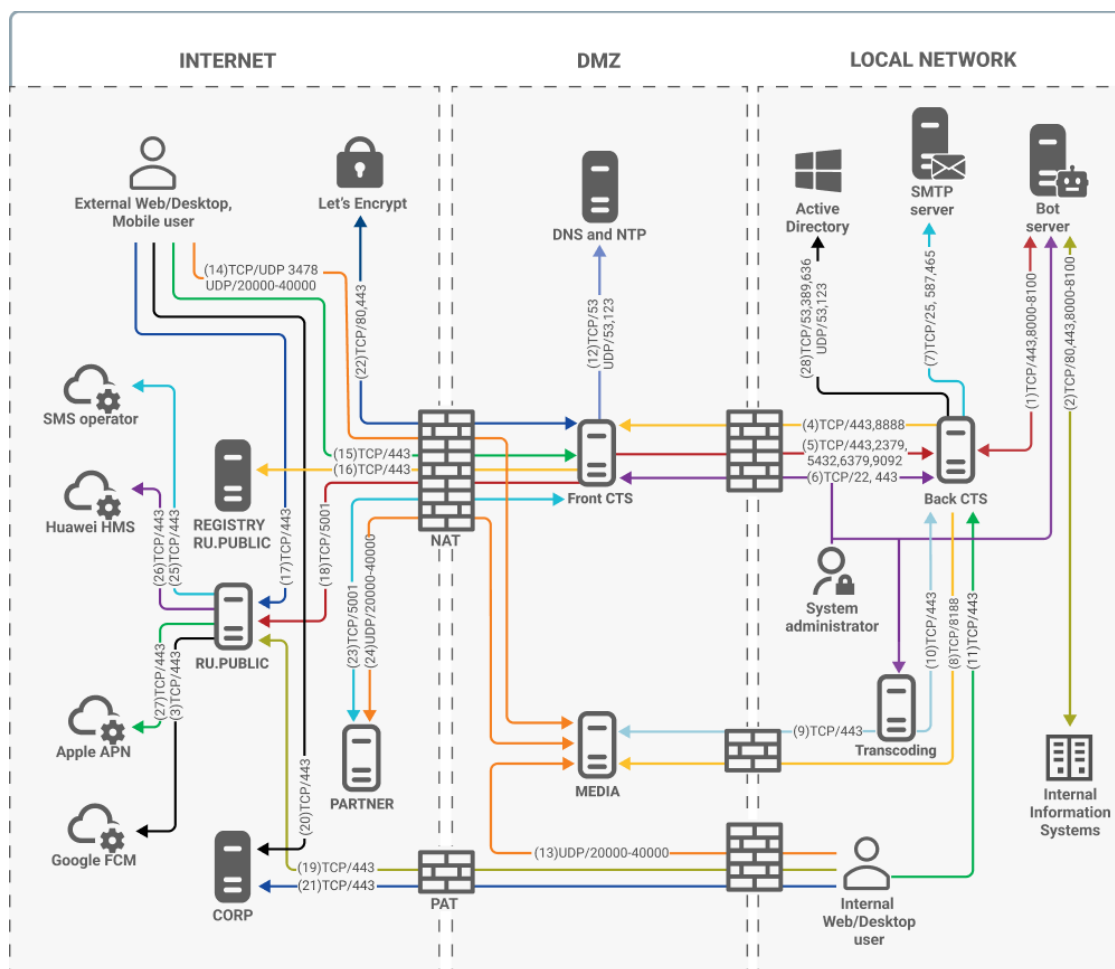


Figure 5

## AUTHENTICATION WITH ADLDS

A server with ADLDS is connected to the CTS server. Synchronization of users from the Active Directory domain controller is performed at regular intervals using a special script. The script is provided by the developer at the deployment stage.

User authentication is performed only by sending a PIN code to the user's e-mail address specified in the user account in Active Directory. When a user account is created, the fields are formed by means of the export of attributes similarly to a direct connection to Active Directory.

A connection example is provided based on a typical corporate server deployment scheme (Figure 6).

The numbers of network interactions correspond to the number of the line in Appendix 2.

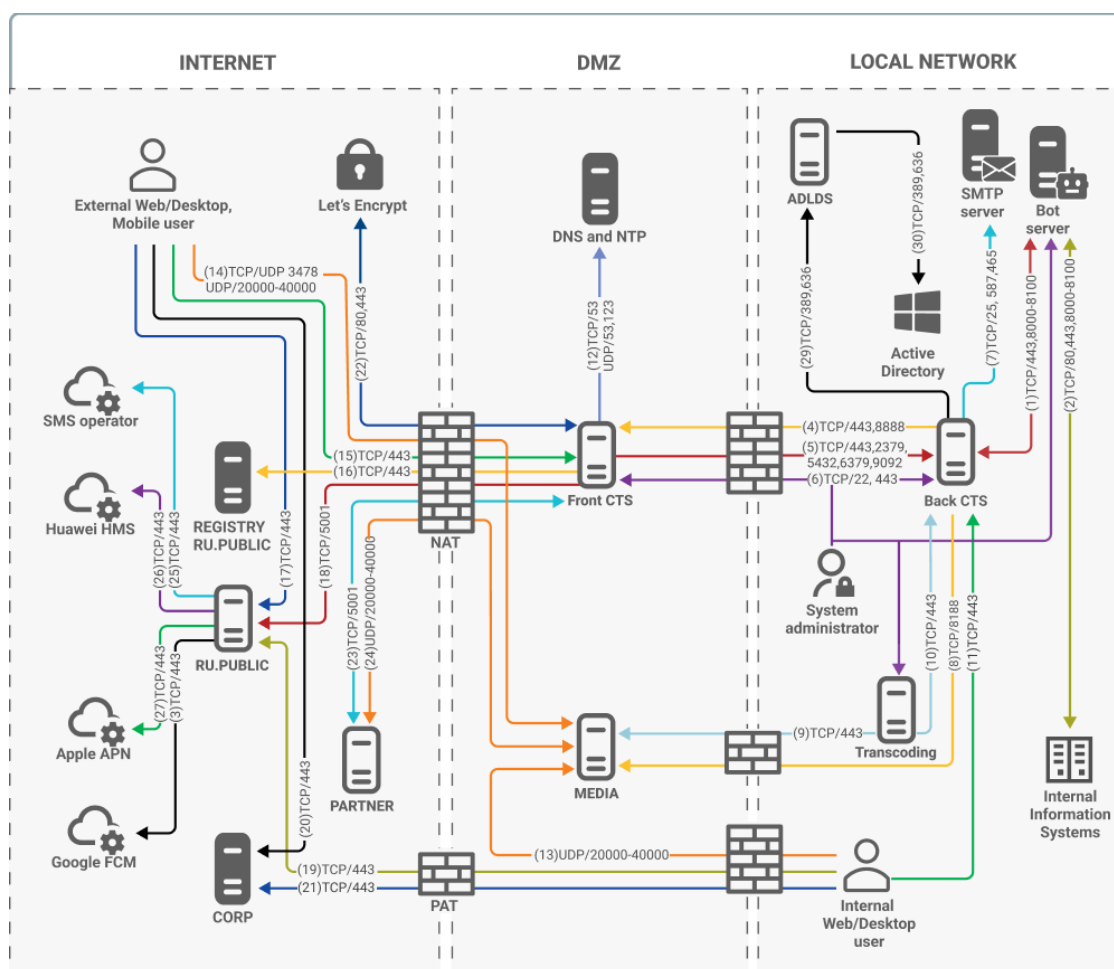


Figure 6

## AUTHENTICATION WITH E-MAIL

User accounts are pre-created on the CTS server manually by the administrator, or the administrator sets up automatic user creation by e-mail mask.

A connection example is provided based on a typical corporate server deployment scheme (Figure 7).

The numbers of network interactions correspond to the number of the line in Appendix 2.

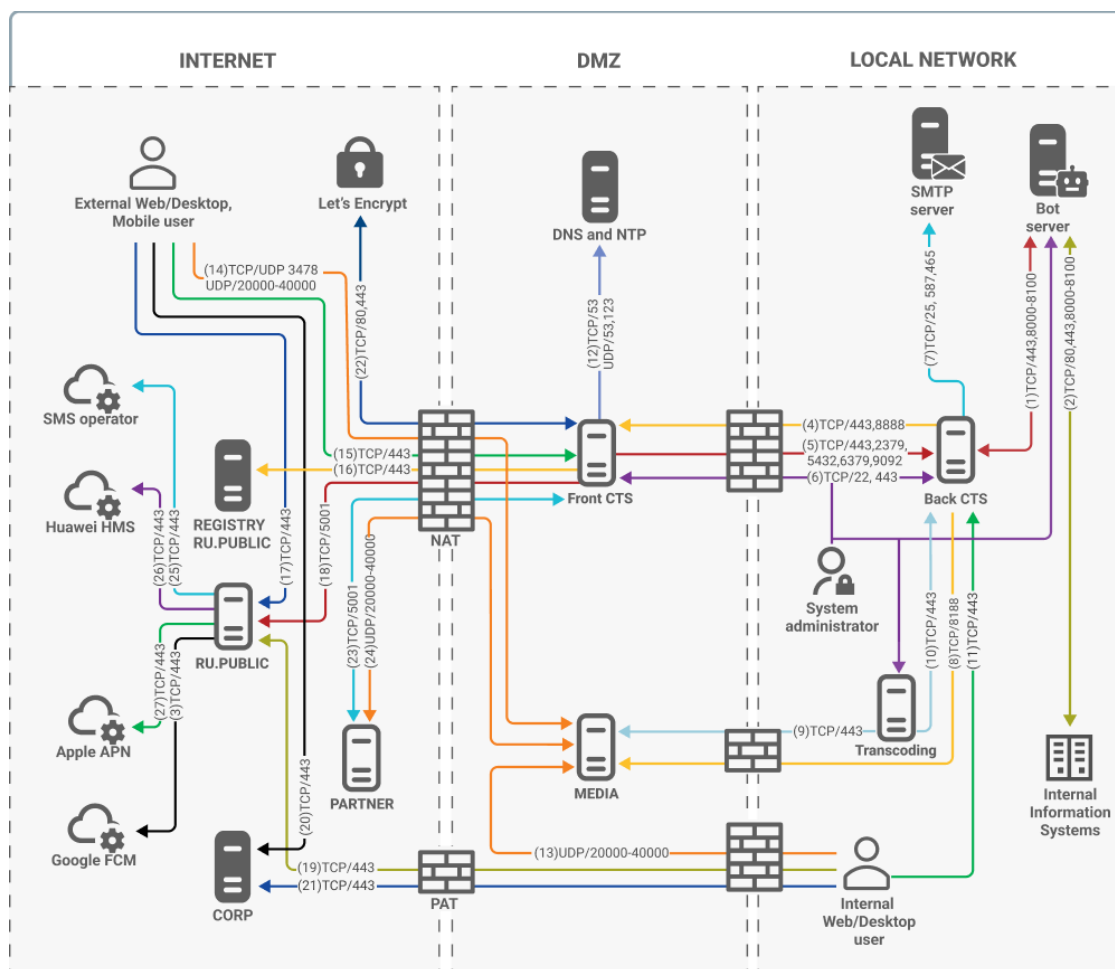


Figure 7

## AUTHENTICATION WITH KEYCLOAK

User data is populated according to account attributes in Keycloak, similar to Active Directory (see [Appendix 9](#) for more information on Keycloak).

The Keycloak service can either connect to the Active Directory controller and other data sources (e.g. HR systems), or maintain its own user base.

A connection example is provided based on a typical corporate server deployment scheme ([Figure 8](#)).

The numbers of network interactions correspond to the number of the line in [Appendix 2](#).

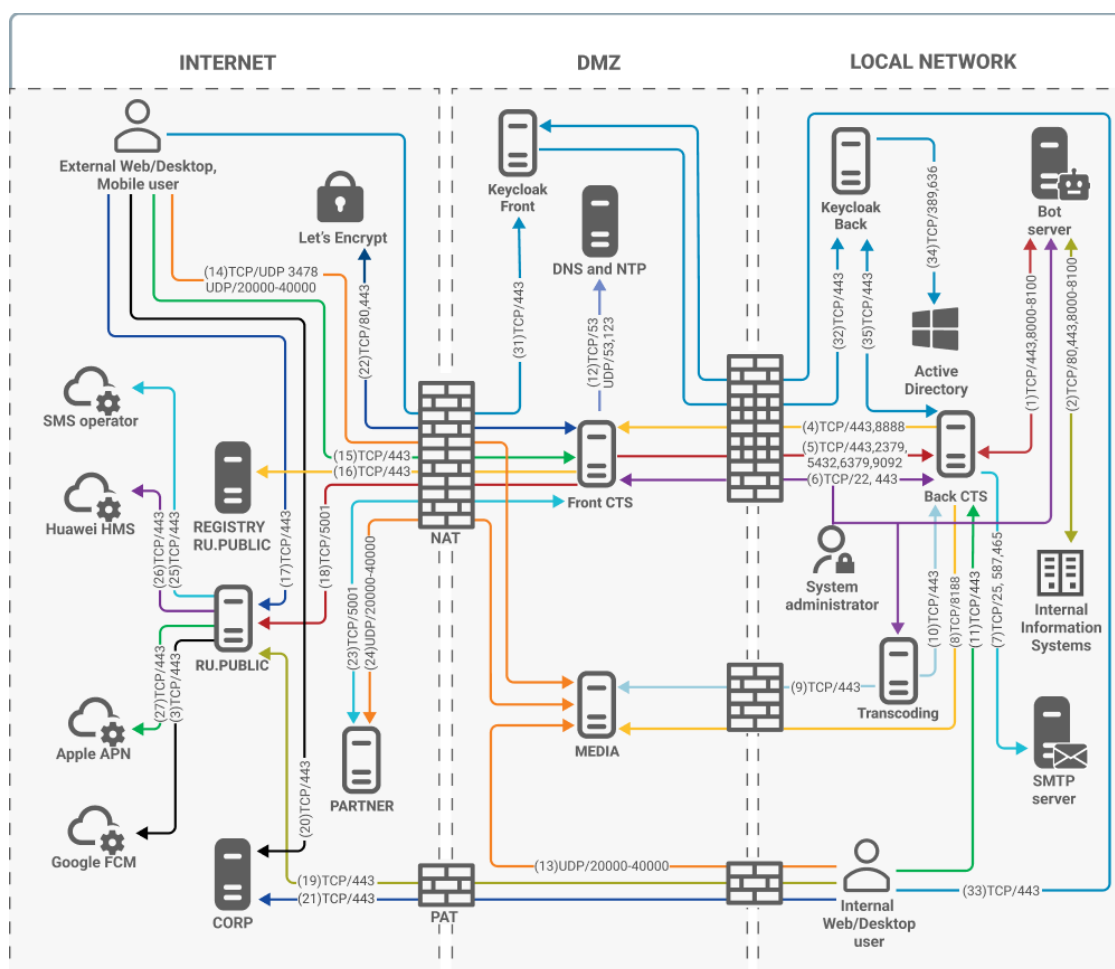


Figure 8

## SYSTEM REQUIREMENTS

### PLATFORM REQUIREMENTS

**Note.** This subsection describes the platform requirements for a non-fault-tolerant configuration based on fewer than 5000 users. If more users are expected, request a custom design from the developer.

The CTS server can be deployed on a hardware platform or in a virtualization environment.

The Front CTS and Media servers shall have one IPv6-enabled network interface (required to run services; IPv6 traffic routing is not required).

Media server calculation is based on an assumption of 0.3 CPU per media call participant and the number of simultaneous participants amounting to 10% of the number of system users. If the number of simultaneous participants in media calls increases, the CPU count should be increased by rounding up to the nearest whole number.

Media server contains a component for processing media call and conference recordings, and requires additional CPU allocation for its operation. The number of allocated CPUs affects the speed of video processing and can be increased in case of low video processing speed. The requirements below take into account the resources needed for the video processing service.

If you plan to have more than 80 concurrent participants in all calls and conferences, you will need to deploy multiple Media servers to reduce the amount of CPU on a single server.

Starting from 100 users and above, a Transcoding server must be allocated separately from the Media server.

Network bandwidth for incoming traffic during video conferencing is calculated as the number of participants multiplied by 1.5 Mbps. Outgoing traffic depends on the call type and the screen layout of the participants. If the participants have a multi-screen (mosaic) layout, then for each participant there is a product of the number of cells in the mosaic (a maximum of 20) and 140 kbps (low quality). The audio channel requires 16 kbps. Screen sharing depends on the nature of the image: for static presentations 30-50 kbps, for dynamic presentations the maximum values can reach 2 Mbps for each participant.

**Important!** To see the minimum system requirements for the Single CTS installation, add together the corresponding parameters for the Front CTS and Back CTS.

*Table 3 Number of users: 100*

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	1	1	45	13
Media	3	2	45	13
Transcoding	2	4	65	13
Back CTS	4	8	211	33
Bot	1	2	65	7
<b>Total</b>	<b>11</b>	<b>17</b>	<b>431</b>	<b>79</b>

*Table 4. Number of users: 200*

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	1	1	45	13
Media	6	4	45	13
Transcoding	2	4	65	13
Back CTS	4	10	358	43
Bot	2	4	85	9
<b>Total</b>	<b>15</b>	<b>23</b>	<b>598</b>	<b>91</b>

*Table 5. Number of users: 300*

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	1	1	45	13
Media	9	4	45	13
Transcoding	3	4	65	13
Back CTS	6	12	504	53
Bot	3	5	105	11
<b>Total</b>	<b>22</b>	<b>26</b>	<b>764</b>	<b>103</b>

*Table 6. Number of users: 400*

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	1	1	45	13
Media	14	8	45	13
Transcoding	4	4	65	13
Back CTS	6	14	651	63
Bot	3	6	100	13
<b>Total</b>	<b>28</b>	<b>33</b>	<b>906</b>	<b>115</b>

Table 7. Number of users: 500

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	2	1	45	13
Media	19	10	45	13
Transcoding	4	4	65	13
Back CTS	8	16	797	73
Bot	4	7	145	15
<b>Total</b>	<b>37</b>	<b>38</b>	<b>1097</b>	<b>127</b>

Table 8. Number of users: 600

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	2	1	45	13
Media	22	12	45	13
Transcoding	4	4	65	13
Back CTS	8	18	944	83
Bot	4	8	165	17
<b>Total</b>	<b>40</b>	<b>43</b>	<b>1264</b>	<b>139</b>

Table 9. Number of users: 700

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	2	1	45	13
Media	25	12	45	13
Transcoding	5	4	65	13
Back CTS	10	18	1090	93
Bot	4	9	185	19
<b>Total</b>	<b>46</b>	<b>44</b>	<b>1430</b>	<b>151</b>

Table 10. Number of users: 800

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	2	2	45	13
Media 1	14	8	45	13
Media 2	14	8	45	13
Transcoding	5	4	65	13
Back CTS	10	20	1237	103
Bot	5	10	205	21
<b>Total</b>	<b>50</b>	<b>52</b>	<b>1642</b>	<b>176</b>

Table 11. Number of users: 900

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	2	2	45	13
Media 1	17	8	45	13
Media 2	17	8	45	13
Transcoding	6	4	65	13
Back CTS	12	22	1383	103
Bot	5	11	225	21
<b>Total</b>	<b>59</b>	<b>55</b>	<b>1808</b>	<b>176</b>

Table 12. Number of users: 1000

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	2	2	45	13
Media 1	18	10	45	13
Media 2	18	10	45	13

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Transcoding	6	4	65	13
Back CTS	12	24	1530	123
Bot	6	12	245	25
<b>Total</b>	<b>62</b>	<b>62</b>	<b>1975</b>	<b>200</b>

Table 13. Number of users: 2000

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	4	2	45	13
Media 1	24	12	45	13
Media 2	24	12	45	13
Media 3	24	12	45	13
Transcoding	12	4	65	13
Back CTS	16	30	2995	223
Bot	7	14	445	45
<b>Total</b>	<b>111</b>	<b>86</b>	<b>3685</b>	<b>333</b>

Table 14. Number of users: 3000

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	6	3	45	13
Media 1	22	12	45	13
Media 2	22	12	45	13
Media 3	22	12	45	13
Media 4	22	12	45	13
Media 5	22	12	45	13
Transcoding	20	4	65	13
Back CTS	20	36	4460	323
Bot	8	16	645	65
<b>Total</b>	<b>164</b>	<b>119</b>	<b>5440</b>	<b>479</b>

Table 15. Number of users: 4000

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	8	4	45	13
Media 1	24	12	45	13
Media 2	24	12	45	13
Media 3	24	12	45	13
Media 4	24	12	45	13
Media 5	24	12	45	13
Media 6	24	12	45	13
Transcoding 1	14	4	65	13
Transcoding 2	14	4	65	13
Back CTS	24	42	5924	423
Bot	10	18	845	85
<b>Total</b>	<b>214</b>	<b>144</b>	<b>7214</b>	<b>625</b>

Table 16 Number of users: 5,000

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Front CTS	10	5	45	13
Media 1	23	12	45	13
Media 2	23	12	45	13
Media 3	23	12	45	13
Media 4	23	12	45	13
Media 5	23	12	45	13

Server Role	vCPU/CPU Core	RAM, GB	SSD, GB	IOps
Media 6	23	12	45	13
Media 7	23	12	45	13
Media 8	23	12	45	13
Transcoding 1	18	4	65	13
Transcoding 2	18	4	65	13
Back CTS	28	48	7364	523
Bot	10	18	1020	105
<b>Total</b>	<b>268</b>	<b>175</b>	<b>8919</b>	<b>771</b>

**Note.** SSD capacity is based on 4 years of log (1 GB) and user data (4 GB) storage depth. Space requirements may differ significantly from design in case of more active product use.

To improve performance and compatibility parameters, as well as to simplify maintenance, it is recommended to use the current versions of system software at the time of installation.

Minimum system requirements for the CTS server for subsystem installation (non-fault tolerant configuration) (see [Table 17](#)):

*Table 17*

Component	Parameters
Processor	The number of cores is selected in accordance with <a href="#">Table 3</a> – <a href="#">Table 16</a> , the frequency shall be not less than 3.60 GHz
RAM	The amount of RAM is selected in accordance with <a href="#">Table 3</a> – <a href="#">Table 16</a>
Operating system	<ul style="list-style-type: none"> <li>• Ubuntu 22.04 LTS or higher;</li> <li>• CentOS 7 or higher;</li> <li>• Centos Stream 8 or higher;</li> <li>• Debian 12.0 or higher;</li> <li>• RHEL 7.1 or higher;</li> <li>• RED OS 7.2 or higher;</li> <li>• Astra Linux Special Edition 1.6 or higher</li> </ul>
Hard drive	Not less than 500 GB
General system software	<ul style="list-style-type: none"> <li>• Docker CE version 20.10.23 or higher;</li> <li>• PostgreSQL version 14 or higher;</li> <li>• etcd version 3.5.x or higher;</li> <li>• Kafka version 2.12 or higher;</li> <li>• Redis version 7.2.4 or higher;</li> </ul>
Network adapter	1 GB/s

Minimum system requirements for the ETS server for subsystem installation (non-fault tolerant configuration) (see [Table 18](#)):

*Table 18*

Component	Parameters
Processor	4 cores, at least 3.60 GHz
RAM	8 GB
Operating system	<ul style="list-style-type: none"> <li>• Ubuntu 22.04 LTS or higher;</li> <li>• CentOS 7 or higher;</li> <li>• Centos Stream 8 or higher;</li> <li>• RHEL 7.1 or higher;</li> <li>• RED OS 7.2 or higher;</li> <li>• Astra Linux Special Edition 1.6 or higher</li> </ul>
Hard drive	Not less than 500 GB
General system software	<ul style="list-style-type: none"> <li>• Docker CE version 20.10.23 or higher;</li> <li>• PostgreSQL version 14 or higher;</li> <li>• etcd version 3.5.x or higher;</li> <li>• Kafka version 2.12 or higher;</li> <li>• Redis version 7.2.4 or higher;</li> </ul>



Component	Parameters
Network adapter	1 GB/s

Minimum system requirements for the RTS server for subsystem installation (non-fault tolerant configuration) (see [Table 19](#)):

*Table 19*

Component	Parameters
Processor	4 cores, at least 3.60 GHz
RAM	16 GB
Operating system	<ul style="list-style-type: none"> <li>• Ubuntu 22.04 LTS or higher;</li> <li>• CentOS 7 or higher;</li> <li>• Centos Stream 8 or higher;</li> <li>• RHEL 7.1 or higher;</li> <li>• RED OS 7.2 or higher;</li> <li>• Astra Linux Special Edition 1.6 or higher</li> </ul>
Hard drive	Not less than 500 GB
General system software	<ul style="list-style-type: none"> <li>• Docker CE version 20.10.23 or higher;</li> <li>• PostgreSQL version 14 or higher;</li> <li>• etcd version 3.5.x or higher;</li> <li>• Kafka version 2.12 or higher;</li> <li>• Redis version 7.2.4 or higher;</li> </ul>
Network adapter	1 GB/s

The eXpress software delivery package includes components for the purpose of demonstrating its functionality. It is not recommended to use them in a production environment. Before installing eXpress software components, it is recommended to develop an architectural installation diagram.

**Note.** The developer of eXpress CS is not responsible for the use of demonstration components in a production environment.

**Operating system requirements:** The CTS, ETC, RTS servers support any Linux family OS on which Docker 20.10.23 can be installed. Ubuntu 20.04 LTS or Ubuntu 18.04 LTS recommended.

**Note.** The CTS, ETC, RTS servers support Astra Linux Orel Common Edition 2.12.43.

**Containerization software requirements:** Docker: 20.10.23 (installation from the docker repository is highly recommended<sup>1</sup>).

**Time synchronization requirement:** An installed and configured local NTP server with a stratum level of at least 15 is required.

It is recommended to use the browsers listed in [Table 20](#) to display web interface.

*Table 20*

Browser	Version
Google Chrome	118
Chromium	120
Yandex Browser	23
Firefox	120

<sup>1</sup> <https://docs.docker.com/install/linux/docker-ce/ubuntu/>

Browser	Version
Opera	100
Edge	118

## DNS REQUIREMENTS

Split DNS technology is used for the proper operation of eXpress CS:

- A DNS name for the CTS server that is resolvable on the Internet and references the external IP address of the Single CTS or Front CTS server publishing address is required. A third-level name such as express.mydomain.tld is recommended;
- In the company's internal network, the DNS name should be resolved to the internal IP address of the CTS server. When using a split installation (Front + Back CTS), each server is assigned an internal DNS name different from the CTS server name.

**Important!** Where impossible to use Split DNS, it is allowed to configure using Linux OS tools (systemd-resolved) with name conversion in the company's internal network to the internal IP-address.

Media server DNS name requirements are similar to those applicable to the DNS name of the CTS server..

## CERTIFICATE REQUIREMENTS

To operate properly, the product requires a certificate for the external name of the eXpress service (FQDN or wildcard), which was issued by a public trusted certification authority and meets the following requirements:

- version 3 and not lower than TLS 1.2;
- key length at least 2048 bits
- SHA 256 signature algorithm;
- X.509 syntax version 3;
- unencrypted private key.

The file must contain the server certificate, intermediate certification authority and root certification authority certificates. The format of the certificates must conform to Base64 encoding. The private key file must contain an unencrypted Base64 encoded private key.

An example certificate file structure is shown in the figure below (see [Figure 9.](#)).

```
-----BEGIN CERTIFICATE-----
Base64 server certificate
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Base64 intermediate ca
-----END CERTIFICATE-----
-----BEGIN CERTIFICATE-----
Base64 root ca
-----END CERTIFICATE-----
```

*Figure 9.*

The use of a free certificate from Let`s Encrypt is supported.

## LDAP CORPORATE DIRECTORY REQUIREMENTS

When integrating eXpress with a Microsoft Active Directory-based corporate directory, it is necessary to create an account with "Domain Users" rights and a "deleted objects" container<sup>1</sup>.

The standard practice of providing user access to eXpress is to create an eXpress user group in Active Directory. The group type is – "Security" and the group visibility is "Universal".

When integrating Express with a corporate directory based on an LDAP-compatible server, it is necessary to create an account with directory read access rights.

When using the AD LDS directory, user authorization is performed exclusively via PIN-code, which is sent to e-mail.

## SMTP SERVER REQUIREMENTS

To be able to send authentication PIN codes to the user's device, it is necessary to create an account on the mail server under which the e-mail will be sent.

## MEDIA SERVER REQUIREMENTS

The Media server can be deployed on a hardware server or in a virtualization environment. The Media server requires a separate FQDN and external IP different from the CTS server.

Minimum system requirements for the Media server are provided in [Table 21](#).

*Table 21*

Component	Parameters
Processor	The number of cores is selected in accordance with <a href="#">Table 3</a> – <a href="#">Table 16</a> , the frequency shall be not less than 3.60 GHz
RAM	The amount of RAM is selected in accordance with <a href="#">Table 3</a> – <a href="#">Table 16</a>
Operating system	<ul style="list-style-type: none"> <li>• Ubuntu 22.04 LTS;</li> <li>• CentOS 7;</li> <li>• Centos Stream 8;</li> <li>• RED OS 7.2 and 7.3;</li> <li>• Astra Linux Special Edition 1.6, 1.7 and 1.8.1</li> </ul>
Hard drive	Not less than 50 GB
General system software	Docker-ce version 20.10.13 or 20.10.23, or 24.0
Network adapter	Ethernet

## NETWORK COMMUNICATION REQUIREMENTS

Networking requirements are described in [Appendix 1](#), [Appendix 2](#), [Appendix 3](#), and [Appendix 4](#).

<sup>1</sup> <https://docs.microsoft.com/ru-ru/troubleshoot/windows-server/identity/non-administrators-view-deleted-object-container>

## WEB CLIENT SERVER REQUIREMENTS

The Web Client server can be deployed on a hardware server or in a virtualization environment. Minimum system requirements for the Web Client server are provided in [Table 22](#).

*Table 22*

Component	Parameters
Processor	2 cores, at least 3.60 GHz
RAM	4 GB
Operating system	<ul style="list-style-type: none"> <li>• Ubuntu 22.04 LTS;</li> <li>• CentOS 7;</li> <li>• Centos Stream 8;</li> <li>• RHEL 7.1, 7.2, 7.3, 7.4, 7.5, 7.6, 7.7, 7.8, 7.9 and 8.0, 8.1, 8.2, 8.3, 8.4, 8.5, 8.6, 8.7, 8.8;</li> <li>• RED OS 7.2 and 7.3;</li> <li>• Astra Linux Special Edition 1.6 and 1.7</li> </ul>
Hard drive	Not less than 50 GB
General system software	Docker CE version 20.10.23
Network adapter	Ethernet

## REQUIREMENTS FOR STORING VIDEOCONFERENCING RECORDING FILES

During the conference recording process, files are created in the highest available quality and then compressed to a resolution of 1920x1080 pixels.

The files which have been successfully created are stored on the CTS server.

Media server serves as a storage for temporary files that are deleted after recording is complete. If the recording was not completed due to failures or errors, the files are stored on Media server for 48 hours.

To store files and ensure a stable recording process, it is necessary to ensure adequate memory capacity on Media server and Single/Back CTS server.

The approximate file size depending on the recording mode is provided in [Table 23](#):

*Table 23*

Recording duration	Description	File size
10 min	Audio recording. Recording of audio from participants' microphones	9.2 MB
10 min	Video broadcast. Recording of video broadcast and audio from participants' microphones	16.4 MB
10 min	Screen sharing. Recording of screen sharing and audio from participants' microphones	53.7 MB

## DLPS REQUIREMENTS

To ensure the operation of DLPS, access to the following objects and functionality is required:

- kafka subsystem to receive "admin-events" and "system-events";
- API of kdc (security key database) and messaging (messaging database) sub-systems;
- messaging and DLP databases;
- LDAP server with LDAP authorization;
- file download.

Network infrastructure requirements for inbound connections (see [Table 24](#)):

*Table 24*

Module/service	Protocol	Port
Web Client	TCP	80, 443

Network infrastructure requirements for outgoing connections (see [Table 25](#)):

*Table 25*

Module/service	Protocol	Port
Kafka	TCP/UDP	9092/9093
Redis	TCP	6379
Postgresql	TCP	5432
CTS-app	TCP	80, 443

Memory requirements (see [Table 26](#)):

*Table 26*

Parameter	Value
Processor	8 cores
RAM	8 GB
Hard drive	40 GB
Network capacity	1 Gbps

# Chapter 2

## INSTALLATION

eXpress installation includes the following steps:

- ETS server deployment:
  1. [Pre-configuration](#).
  2. [Enterprise server installation](#).
  3. [Web Client installation](#) (optional).
  4. [Media pre-configuration](#).
  5. [Media server installation](#).
  6. [Transcoding server installation](#) (optional).
  7. [Corporate server installation](#).
  8. [Media server connection to the corporate server](#).
  9. [Media server setup](#).
  10. [Link service installation](#) (optional).
  11. [DLPS installation](#) (optional).
  12. [Installation of call and conference recording components](#) (optional).
  13. [Checking certificates](#) (optional).
  14. [Server launch](#).
  15. [Server setup](#).
- CTS server deployment:
  1. [Pre-configuration](#).
  2. [Media pre-configuration](#).
  3. [Media server installation](#).
  4. [Transcoding server installation](#) (optional).
  5. [Corporate server installation](#).
  6. [Media server connection to the corporate server](#).
  7. [Media server setup](#).
  8. [Link service installation](#) (optional).
  9. [DLPS installation](#) (optional).
  10. [Installation of call and conference recording components](#) (optional).
  11. [Checking certificates](#) (optional).
  12. [Server launch](#).
  13. [Server setup](#).

## PRE-CONFIGURATION

To ensure correct operation of the server, it is necessary to perform its pre-configuration.

---

**Attention!** eXpress installation shall be performed by a Linux user with administrative experience.

---

Pre-configuration process depends on the OS.

## UBUNTU/DEBIAN OS

**To perform pre-configuration when using Ubuntu/Debian OS:**

1. Install Ubuntu 22.04 LTS or Ubuntu 20.04 LTS operating system. Use the official source to install the distribution package:

<https://ubuntu.com/download/server>

**Attention!** During operating system installation, allocate 24 GB for the root "/" partition, disable the SWAP partition, and allocate the remaining space for the "/var/lib/docker" partition.

2. Delete the snapd and ufw packages using the following command:

```
apt autoremove --purge snapd ufw
```

3. Install Docker software. To install, use the official source:

<https://docs.docker.com/install/linux/docker-ce/ubuntu/>

**Attention!** If Docker software is extracted from the snapd package, remove it and install it from the official source.

Sample code to install Docker:

```
#Uninstall all conflicting packages
for pkg in docker.io docker-doc docker-compose docker-compose-v2
podman-docker containerd runc; do sudo apt-get remove $pkg; done

# Add Docker's official GPG key:
sudo apt-get update
sudo apt-get install ca-certificates curl
sudo install -m 0755 -d /etc/apt/keyrings
sudo curl -fsSL https://download.docker.com/linux/ubuntu/gpg -o
/etc/apt/keyrings/docker.asc
sudo chmod a+r /etc/apt/keyrings/docker.asc

# Add the repository to Apt sources:
echo \
  "deb [arch=$(dpkg --print-architecture) signed-
by=/etc/apt/keyrings/docker.asc]
https://download.docker.com/linux/ubuntu \
  $(. /etc/os-release && echo "$VERSION_CODENAME") stable" | \
  sudo tee /etc/apt/sources.list.d/docker.list > /dev/null
sudo apt-get update

#Install the latest version
sudo apt-get install docker-ce docker-ce-cli containerd.io docker-
buildx-plugin docker-compose-plugin
```

4. Install additional software (see below).

**To install additional software:**

1. Install the NTP server using the following command:

```
apt install chrony
```

If there are sources of exact time within the company, specify the <sup>1</sup>NTP servers in the file /etc/chrony/chrony.conf as follows:

```
server ntp1.local
```

<sup>1</sup> This refers to customer servers that use NTP servers.

```
server ntp2.local
server ntp3.local
```

Sample code:

```
systemctl enable chrony
systemctl restart chrony
```

To test connection to NTP servers, use the following command:

```
chronyc sources -v
```

- Specify Docker log storage parameters in the `/etc/docker/daemon.json` directory as follows:

```
{
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  }
}
```

- Run the following command:  

```
systemctl restart docker
```
- Check the SSL certificate chain and ensure that the certificates are in the correct order (see page 34).
- Check that the server settings<sup>1</sup> are correct [Table 27](#):

*Table 27*

Setting name	Definition	Purpose
Open CTS server ports	22, TCP	Remote connection to SSH for server management
Open DTS server ports	53, UDP/TCP	DNS queries
Open NTP server ports	123, UDP	Time synchronization via the NTP protocol
Open AD server ports	389, TCP	Connecting to the AD server for user authorization and obtaining the list of users
Open AD server ports	636, TCP	TLS connection to the AD server for authorization and obtaining the list of users
Open CTS server ports	443, TCP	HTTPS connection of mobile clients to the CTS server
Open registry.public.express server ports	443, TCP	Installing and updating CTS packages
Open port ru.public.express:5001	5001, TCP	Trust connection to the Russian regional server
Open Media server port	8188, TCP	Connecting to Janus on Media server
DNS name	<ul style="list-style-type: none"> <li>It is recommended to have a third DNS level;</li> <li>In the company's internal network, the DNS name should be resolved to the internal IP address of Single CTS or Back CTS;</li> <li>Media server DNS name requirements are similar to</li> </ul>	

<sup>1</sup> These settings are suitable for installing all components on two servers. For detailed settings of network interactions for Single CTS and combination of Front CTS and Back CTS, see page 45, [Appendix 2](#) and page 46 [Appendix 3](#), respectively.



Setting name	Definition	Purpose
	those applicable to the DNS name of the CTS server.	
DNS name certificate	<ul style="list-style-type: none"> <li>• SSL version 3 and not lower than TLS 1.2;</li> <li>• key length of 2,048 or greater;</li> <li>• X.509 version 3;</li> <li>• unencrypted certificate key<sup>1</sup></li> </ul>	
Microsoft AD account	Active account with read access to the selected group and deleted objects	Obtaining the list of users

6. Request from the developer the following individual parameters for installation (parameters are provided by the FQDN of the specific server) (see Table 28):

Table 28

Parameter	Description
cts_id	This server ID;
rts_host	FQDN address of the RTS server to which the CTS server will be connected;
rts_id	Identification of the RTS Server;
rts_token	Token for authorization on the RTS server. Has the following format <token_for_accept>:<token_for_connect>, where token_for_accept is the token to accept connection from the remote server, token_for_connect is the token to connect to the remote server

## CENTOS/RHEL OPERATING SYSTEM

### To perform pre-configuration when using Centos/RHEL operating system:

1. Install Centos/RHEL OS operating system.
2. Remove firewalld using the following command:

```
systemctl disable firewalld
```

or:

```
systemctl stop firewalld
```

3. Set SELinux to Permissive mode by editing the /etc/selinux/config file.
4. Install Docker software. To install, use the official source<sup>2</sup>.
5. Install the NTP server (see below).

### To install the NTP server:

1. Install the NTP server using the following command:

```
dnf install chrony
```

2. If there are sources of exact time within the company, specify the <sup>3</sup>NTP servers in the file /etc/chrony.conf as follows:

```
server ntp1.local
server ntp2.local
```

<sup>1</sup> Can be provided by the developer company.

<sup>2</sup> <https://docs.docker.com/engine/install/centos/>

<sup>3</sup> This refers to customer servers that use NTP servers.

```
server ntp3.local
```

Sample code:

```
systemctl enable chrony
systemctl start chrony
```

**To test connection to NTP servers**, use the following command:

```
chronyc sources -v
```

---

## ASTRA LINUX EAGLE OPERATING SYSTEM

### To perform pre-configuration when using Astra Linux Eagle OS:

1. Install Astra Linux Eagle operating system. During installation, in the Select Software selection step, select Basic Tools, SSH Remote Access Tools.
2. Install Docker using the following command:

```
apt install docker.io
```

3. Install additional software (see below).

### To install additional software:

1. Install the NTP server using the following command:

```
apt install chrony
```

If there are sources of exact time within the company, specify the <sup>1</sup>NTP servers in the file `/etc/chrony/chrony.conf` as follows:

Remove or comment out the pool line and specify your servers.

Example:

```
server ntp1.local
server ntp2.local
server ntp3.local
```

2. Restart the service to apply the changes:

```
systemctl restart chrony
```

To test connection to NTP servers, use the following command:

```
chronyc sources -v
```

3. Get root privileges using the following command:

```
sudo -s
```

4. Specify Docker log storage parameters in the `/etc/docker/daemon.json` directory as follows:

```
{
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  }
}
```

5. Run the following command:

```
systemctl restart docker
```

6. Check the SSL certificate chain and ensure that the certificates are in the correct order (see page [34](#)).

---

<sup>1</sup> This refers to customer servers that use NTP servers.

7. Check that the server settings<sup>1</sup> are correct [Table 29](#):

Table 29

Setting name	Definition	Purpose
Open CTS server ports	22, TCP	Remote connection to SSH for server management
Open DTS server ports	53, UDP/TCP	DNS queries
Open NTP server ports	123, UDP	Time synchronization via the NTP protocol
Open AD server ports	389, TCP	Connecting to the AD server for user authorization and obtaining the list of users
Open AD server ports	636, TCP	TLS connection to the AD server for authorization and obtaining the list of users
Open CTS server ports	443, TCP	HTTPS connection of mobile clients to the CTS server
Open registry.public.express server ports	443, TCP	Installing and updating CTS packages
Open Media server port	8188, TCP	Connecting to Janus on the Media server
Open port ru.public.express:5001	5001, TCP	Trust connection to the Russian regional server
DNS name	<ul style="list-style-type: none"> <li>It is recommended to have a third DNS level;</li> <li>In the company's internal network, the DNS name should be resolved to the internal IP address of Single CTS or Back CTS;</li> <li>Media server DNS name requirements are similar to those applicable to the DNS name of the CTS server.</li> </ul>	
DNS name certificate	<ul style="list-style-type: none"> <li>SSL version 3 and not lower than TLS 1.2;</li> <li>key length of 2,048 or greater;</li> <li>X.509 version 3;</li> <li>unencrypted certificate key<sup>2</sup></li> </ul>	
Microsoft AD account	Active account with read access to the selected group and deleted objects	Obtaining the list of users

8. Request from the developer the following individual parameters for installation (parameters are provided by the FQDN of the specific server) (see [Table 30](#)):

Table 30

Parameter	Description
cts_id	This server ID;
rts_host	FQDN address of the RTS server to which the CTS server will be connected;
rts_id	Identification of the RTS Server;

<sup>1</sup> These settings are suitable for installing all components on two servers. For detailed settings of network interactions for Single CTS and combination of Front CTS and Back CTS, see page 45, [Appendix 2](#) and page 46 [Appendix 3](#), respectively.

<sup>2</sup> Can be provided by the developer company.

Parameter	Description
rts_token	Token for authorization on the RTS server. Has the following format <token_for_accept>:<token_for_connect>, where token_for_accept is the token to accept connection from the remote server, token_for_connect is the token to connect to the remote server

## INSTALLING ETS

The following set of commands shall be run in the command line of the server on which the ETS server is installed.

### To install the ETS server:

1. Open the Command Prompt.
2. Connect to the developer's Docker repository to download containers:

```
docker login -u Login -p Password registry.public.express
```

**Note.** Login and Password, which are issued by the developer, are used as login and password.

3. Download the container installer:

```
docker run --rm registry.public.express/dpl:ets-release dpl-install | bash
```

A YAML file with containers and an installer will be downloaded from the repository to the server.

4. Create a working directory for the ETS server:

```
mkdir -p /opt/express
cd /opt/express
echo DPL_IMAGE_TAG=ets-release > dpl.env
dpl --init
```

After running the dpl --init command, a settings.yaml file is created.

5. Install SSL certificate and key chains:

- When using own certificate, create a directory for the certificates.

**Attention!** The certificate file name and key name should match the example below:

```
mkdir -p certs
cp /somewhere/my-certificate-chain.crt certs/express.crt
cp /somewhere/my-unencrypted-key.key certs/express.key
```

The /somewhere/my-certificate-chain.crt and /somewhere/my-unencrypted-key.key constructs are individual for each specific case.

The certs/express.crt and certs/express.key constructs are mandatory.

Requirements for certificates are set out on page 34;

- when using a Let's Encrypt certificate, add the parameter le\_email: [admin@company-mail.ru](mailto:admin@company-mail.ru) to the settings.yaml file.

Checking the connection of certificates after installation is described on page 70.

6. Configure DLPS to allow security administrators to access message content (for setup options, see page 67).
7. Install cAdvisor (installation is performed from the /opt/express directory):

```
dpl cadvinstall
ps ax|grep cadvisor | grep -v grep
```

Command output:

```
17605 ? Ssl 44:20 /usr/bin/cadvisor -listen_ip 172.17.0.1 -port 9100
```

8. Install Prometheus node exporter from the /opt/express directory using the following command:

```
dpl nxinstall
ps ax|grep node_exporter | grep -v grep
```

Command output:

```
17802 ? Ssl 322:51 /usr/bin/node_exporter --web.listen-address=172.17.0.1:9200
```

Once the installation of the ETS server and supporting software is complete, a configuration file is created in which you need to set parameters for connecting to the RTS server, receiving push notifications, SMS messages and other functions.

The configuration file created by default looks like this and requires editing:

```
api_internal_token:
ccs_host: cts_name.somedomain.sometld
ets_id: 'dddd-cccc-dddd-cccc'
phoenix_secret_key_base:
postgres_password:
prometheus_users:
  prometheus:
rts_host: 'rts_name.somedomain.sometld'
rts_id: 'aaaa-bbbb-cccc-dddd'
rts_token: 'verystrongpassword'
```

**To change the configuration file**, use any text editor and make the following corrections to the file. The list of all settings in the configuration file is provided in [Table 31](#).

Table 31

Setting name	Value
<b>Mandatory settings</b>	
ccs_host	The full domain name of this server, which is registered in DNS and the corresponding name for which the certificate was purchased
ets_id	ID of the installed server, provided by the developer
prometheus_users	List of users with passwords generated by the htpasswd utility to access the Prometheus stack integrated into the system
rts_host	Full domain name of the RTS server to which the installed CTS server will connect (provided by the developer)
rts_id	ID of the RTS Server (provided by the developer)
rts_token	Token for authorization on the RTS server (provided by the developer)
le_email	This parameter is set when using a certificate from Let's Encrypt. The value of the parameter shall correspond to the e-mail to which notifications from Let's Encrypt will be sent
admin_url	This parameter is specified to override the standard path (/admin) to the administrator web interface: for example /not-admin
<b>Optional settings</b>	
Access to administrator web interface and DLPS administrator console	admin_allow: <ul style="list-style-type: none"> <li>- 10.0.0.0/8</li> <li>- 172.16.1.0/24</li> </ul>
Access to Prometheus	prometheus_allow: <ul style="list-style-type: none"> <li>- 10.0.0.0/8</li> <li>- 172.16.1.0/24</li> </ul>
Changing the path to the administrator interface	admin_url: /express-admin-ui
Changing the path to the	dlps_url: /dlps-admin-ui

Setting name	Value
DLPS administrator interface	

It is recommended to fix the `ets_id`, `rts_host`, `rts_id` and `rts_token` parameters **for the server to function correctly**; they are highlighted in red in the example above.

**Note.** The values of the `ets_id`, `rts_host`, `rts_id` and `rts_token` parameters must be inside quotation marks ('value'). This requirement does not apply to other parameters. To prevent errors, it is recommended to replace the parameters of the generated file with the parameters issued by the developers. When entering values manually, the quote characters are not entered.

For this type of architecture, [install a web client](#).

## WEB CLIENT INSTALLATION

**Attention!** Web Client is installed on the server after installing Docker CE and Docker Compose.

The Web Client connects to the ETS and CTS servers. The Web Client can be installed at any time, but login will only be possible after the CTS server is installed.

### To install Web Client:

1. Open the Command Prompt.
2. Connect to the developer's Docker repository to download containers:

```
docker run --rm registry.public.express/dpl:web-release dpl-
install | bash
docker login -u Login -p Password registry.public.express
```

**Note.** Login and Password, which are issued by the developer, are used as login and password.

3. Create a working directory for Web Client:

```
mkdir -p /opt/web_client
cd /opt/web_client
echo DPL_IMAGE_TAG=web-release > dpl.env
dpl --init
```

4. After running the `dpl --init` command, a `settings.yaml` file is created.
5. Install SSL certificate and key chains:

- When using own certificate, create a directory for the certificates.

**Important!** The certificate file name and key name should match the example below:

```
mkdir -p certs
cp /somewhere/my-certificate-chain.crt certs/express.crt
cp /somewhere/my-unencrypted-key.key certs/express.key
```

The `/somewhere/my-certificate-chain.crt` and `/somewhere/my-unencrypted-key.key` constructs are individual for each specific case.

The `certs/express.crt` and `certs/express.key` constructs are mandatory.

Requirements for certificates are set out on page [34](#):

- when using a Let's Encrypt certificate, add the parameter `le_email`: [admin@company-mail.ru](mailto:admin@company-mail.ru) to the `settings.yaml` file.

Checking the connection of certificates after installation is described on page 70.

6. The configuration file created by default looks like this and requires editing:

```
ccs_host: somehost.somedomain.sometld
web_client_config: ''
```

7. Example of configuration settings:

```
ccs_host: example.com
le_email: test@example.com
web_client_enabled: true
web_client_config:
  regions:
    ru:
      host: rts1dev.server.ru
      prefix: 7
    ae:
      host: rts2dev.server.ru
      prefix: 971
sentryDSN: https://sentryToken@sentry.server.ru/58
ccsHost: corp.express
ctsWeb: false
locales: ["en","ru","de","fr","es"]
platformPackageId: ru.unlimitedtech.express
gcmSenderId: senderId
landingUrl: https://express.ms/mobile-corp-express
allowCtsLogin: true
allowDebugInfo: true
ets: true
gmapsApiKey: apiKeyapiKeyapiKey
actionTaskFeature: true
changelogUrl: https://dl.express.ms/changelog/changelog-{}.md
images:
  web_client: registry.public.express/web_client:develop
```

8. In the /opt/express/web\_client directory, run the following command:

```
dpl -d
```

## MEDIA SERVER INSTALLATION

Media server is designed to organize video and audio communications between users. The video uses the default VP8 codec, 120 kbps, 360 kbps, 1,080 kbps bitrate per participant (depending on the selected quality on the client side). Audio uses the default OPUS codec, 16 kbps bit rate per participant. The total bandwidth that a client can use does not exceed 2,500 kbps.

Installation of the Media server is carried out in the following order:

- [pre-configuration](#);
- [Media server installation](#);
- [installation of a corporate server \(Single CTS or Front CTS and Back CTS servers\)](#);
- [connecting Media server to CTS](#);
- [setting up Media server](#).

## PRE-CONFIGURATION

To ensure correct operation of the server, it is necessary to perform its pre-configuration.

**Note.** Delays in the transmission of voice information in TURN mode depend on the distance of the end user from the TURN server.

### Prior to Media server installation:

1. Determine the global IP address for the Media server that is accessible from the Internet.
2. Check that the server settings are correct in accordance with [Table 32](#):

*Table 32*

Direction	Source	Receiver	Port	Protocol	Purpose of the port
Incoming	Admin IP	Media	22	TCP	SSH
Incoming	CTS	Media	8188	TCP	Management conference
Incoming	Any	Media	3478	TCP/UDP	TURN
Incoming	Any	Media	20000-40000	UDP	SRTP media
Outgoing	Media	Any	Any	UDP	SRTP media
Outgoing	Media	DNS	53	TCP/UDP	DNS
Outgoing	Media	NTP server	123	UDP	NTP server
Outgoing	Media	registry.public.express	443	TCP	Docker registry

3. Assign a domain name to the Media server.
4. Prepare the SSL certificate chain in the PEM format and an unencrypted private key.

## MEDIA SERVER INSTALLATION

The following set of commands is shall be run in the command line of the server on which the Media server is installed.

### To install Media server:

1. Connect to the Media server via SSH.
2. Open the Command Prompt.
3. Install the NTP server using the following command:

```
apt install chrony
```

If there are sources of exact time within the company, specify the NTP servers in the file `/etc/chrony/chrony.conf` as follows:

```
server ntp1.local
server ntp2.local
server ntp3.local
```

Sample code:

```
systemctl enable chrony
systemctl restart chrony
```

To test connection to NTP servers, use the following command:

```
chronyc sources -v
```

4. Connect to the developer's Docker repository to download containers:

```
docker login -u Login -p Password registry.public.express
```



**Note.** Login and Password, which are issued by the developer, are used as login and password.

5. Download the container installer:

```
docker run --rm registry.public.express/dpl:voex-release dpl-
install | bash
```

A YAML file with containers and an installer will be downloaded from the repository to the server.

6. Create a working directory for the project:

```
mkdir -p /opt/express-voice
cd /opt/express-voice
echo DPL_IMAGE_TAG=voex-release > dpl.env
dpl --init
```

7. Install the certificate chain and SSL key for the TURN and STUN servers:

```
mkdir -p certs
cp /somewhere/my-certificate-chain.crt certs/express.crt
cp /somewhere/my-unencrypted-key.key certs/express.key
```

8. Create the DH (Diffie Hellman) key:

```
openssl dhparam -out certs/dhparam.pem 2048
```

9. Open the /opt/express-voice/settings.yaml file for editing:

```
external_interface: eth0
janus_ws_enable: true
janus_ws_ip: 127.0.0.1
janus_wss_enable: false
janus_wss_ip: 127.0.0.1
- ' '
ccs_host: somehost.somedomain.sometld
phoenix_secret_key_base: *****
turnserver_server_name: localhost
turnserver_listening_ip: 127.0.0.1
api_internal_token: token
```

Default settings are described in [Table 33](#):

*Table 33*

Setting name	Value
external_interface	Name of the interface with external IP address
janus_keep_private_host	Enabling connection negotiation for all local IP addresses of the server
ccs_host	FQDN name of the Media server
api_internal_token	Token for API queries
janus_ws_acl	The addresses or networks of servers where the messaging container is located (for example, 172.18.0.)
janus_ws_ip - ip	An interface that uses janus websocket to manage conferences with the messaging service
janus_wss_enable janus secure websocket	Enabling janus secure websocket
janus_wss_ip	An interface that uses janus secure websocket
nat_1_1_mapping keep_private_host	When using NAT 1:1, the external IP address is specified and the private IP address saving mode is enabled
keep_private_host	The list of allowed IP addresses: <ul style="list-style-type: none"> <li>• for one CTS server – its address: [1.2.3.4];</li> <li>• if the CTS and Media servers are on the same server – empty list: []</li> </ul>
phoenix_secret_key_base	Server key (leave unchanged)
turnserver_shared_key	Key to connect Media to the CTS server (copy and save the generated key value)

Setting name	Value
turnserver_external_ip	External IP address
turnserver_listening_ip	External or internal IP address of the interface for TURN and STUN servers
transcoding_storage_enabled	Enable temporary storage of records service, disabled by default

- Make changes to the default settings and add the following parameter:

```
turnserver_external_ip:
- 1.2.3.4
```

- Use the command below to generate the value of turnserver\_shared\_key:

```
cat /proc/sys/kernel/random/uuid | tr -d '-' | base64 | cut -b 1-22
```

- Copy and save the generated key value (YmNjY2VmNDk0ZTEwNDgzNj is used as an example) to further [connect the Media server to the CTS server](#).

- Add this parameter to the configuration:

```
turnserver_shared_key: YmNjY2VmNDk0ZTEwNDgzNj
```

- If call recording is to be used, add the parameter:

```
transcoding_storage_enabled: true
```

- Add the following parameters and set the parameter "janus\_nat\_1\_1\_mapping" equal to the value of the external IP address on the Internet from which port forwarding is performed:

```
janus_keep_private_host: true
janus_ws_ip: 172.17.0.1
janus_ws_acl: 172.19.0.
janus_nat_1_1_mapping: 1.2.3.4
```

- Run the command to pre-generate configuration files:

```
dpl -p
```

- Run the following command:

```
dpl -d
```

If the architectural solution involves decoupled installation of the conference recording service (Recording Bot) and the conference service (Media), please contact the developer company.

## TRANSCODING SERVER INSTALLATION

### To install and configure the Transcoding server:

- Connect to the dedicated server via SSH.
- Create a folder for transcoding to work with:

```
mkdir -p /opt/transcoding
```

- Install the Docker service:

```
curl -fsSL http://get.docker.com -o get-docker.sh && sh get-docker.sh
```

- Specify Docker log storage parameters in the /etc/docker/daemon.json directory as follows:

```
{
  "log-driver": "json-file",
  "log-opts": {
    "max-size": "100m"
  }
}
```

```
}
```

5. Restart the Docker service:

```
systemctl restart docker
```

6. Go to the following directory: /opt/transcoding:

```
cd /opt/transcoding
```

7. Create a project variable:

```
echo "DPL_IMAGE_TAG=voex-release" > dpl.env
```

8. Create and run a Docker container:

```
docker run --rm registry.public.express/dpl:voex-release dpl-  
install | bash
```

9. Initialize the VoEx project:

```
dpl --init
```

10. Open the file /opt/transcoding/settings.yaml in any text editor (for example, nano):

```
nano /opt/transcoding/settings.yaml
```

11. Disable the redis, coturn and janus services by setting them to false:

```
coturn_enabled: false  
janus_enabled: false  
redis_enabled: false  
transcoding_storage_enabled: false
```

12. Add hosts for transcoding to work. The parameters for transcoding are described in [Table 34](#). The values can be copied from the Media Server, from which the container is migrated.

**Important!** Copy the `api_internal_token` values from the `/opt/express/settings.yaml` files located on the corresponding `ccs_hosts` servers. Copy the token values from the `/opt/express-voice/settings.yaml` files (`api_internal_token` value) located on the corresponding Media servers.

Example of transcoding host settings:

- for a single CTS server:

```
transcoding_hosts:  
  cts:  
    ccs_host: fqdn_cts  
    api_internal_token: $api_internal_token_cts  
    storages_tokens_mapping:  
      fqdn_medial:  
        token: $api_internal_token_media  
        ssl_envs_prefix: "TSS"  
# optional parameters  
# if the certificates are not public, then disable certificate  
verification:  
tc-cts_env_override:  
  TSS_SSL_ENABLED: true  
  TSS_SSL_VERIFY: verify_none
```

- for multiple CTS servers:

```
transcoding_hosts:  
  cts1:  
    ccs_host: fqdn_cts1  
    api_internal_token: $api_internal_token_cts1  
    storages_tokens_mapping:  
      fqdn_medial:  
        token: $api_internal_token_medial  
        ssl_envs_prefix: "TSS"  
  cts2:
```

```

ccs_host: fqdn_cts2
api_internal_token: $api_internal_token_cts2
storages_tokens_mapping:
  fqdn_media2:
    token: $api_internal_token_media2
  ssl_envs_prefix: "TSS"

```

Table 34

Setting name	Value
transcoding_hosts	<p>The list of hosts objects (CTS) consists of the following parameters:</p> <ul style="list-style-type: none"> <li><b>cts (cts1, cts2)</b> — unique name, fqdn_cts can be used;</li> <li><b>ccs_host</b> — FQDN of the CTS server;</li> <li><b>api_internal_token</b> — token for API queries (copy from the /opt/express/settings.yaml files located on the corresponding ccs_hosts servers).</li> </ul> <p>It can contain multiple <b>cts</b> blocks if you have one transcoding server for multiple CTS servers</p>
storages_tokens_mapping	<p>The list of hosts objects consists of the following parameters:</p> <ul style="list-style-type: none"> <li><b>fqdn_media</b> — FQDN of the Media server, must be unique;</li> <li><b>token</b> — api_internal_token of the Media server;</li> <li><b>ssl_envs_prefix</b> — certificate prefix.</li> </ul> <p>May contain multiple <b>fqdn_media</b> blocks if the CTS server has more than one Janus(janus_ws_url)</p>
tc-ct_env_override	Additional parameters for transcoding
TSS_SSL_ENABLE	Enable/disable advanced transcoding settings
TSS_SSL_VERIFY	Verification of certificate for transcoding

13. Start the service using the following command:

```
dp1 -d
```

14. Check the status of containers using the following command:

```
docker ps -a
```

The command should result in transcoding containers appearing according to the value specified in the transcoding\_hosts variable, for example:

```

root@yc-msg-ext-voex-transcoding01:~# docker ps -a
CONTAINER ID   IMAGE
COMMAND        CREATED        STATUS        PORTS
NAMES
dd5ca4e7bdee   registry.public.express/transcoding:3.24.0
"/bin/sh -c 'export ..." 45 hours ago  Up 22 hours  4000/tcp
voex-tc-cts-1

```

15. Check the availability of the Media Server using the following command:

```
curl https://fqdn-media/testtest
```

16. To get the Docker container logs on the Media server, run the following command:

```
docker logs voex-nginx-1 | grep testtest
```

The response must contain the request(s):

```

voice.test.corp.express 172.18.0.2 - - [02/Oct/2024:08:50:34
+0000] "GET /testtest HTTP/1.1" 204 0 "-" "curl/8.5.0"
"51.250.102.111"

```

## CORPORATE SERVER INSTALLATION

**Important!** Before starting the installation procedure, it is necessary to install the Media server (see page 44).

## SINGLE CTS INSTALLATION

The following set of commands shall be run in the command line of the server on which the CTS server is installed.

**To install the CTS server:**

1. Open the Command Prompt.
2. Connect to the developer's Docker repository to download containers:

```
docker login -u Login -p Password registry.public.express
```

**Note.** Login and Password, which are issued by the developer, are used as login and password.

3. Download the container installer:

```
docker run --rm registry.public.express/dpl:cts-release dpl-install | bash
```

A YAML file with containers and an installer will be downloaded from the repository to the server.

4. Create a working directory for the CTS server:

```
mkdir -p /opt/express
cd /opt/express
echo DPL_IMAGE_TAG=cts-release > dpl.env
dpl --init
```

After running the dpl --init command, a settings.yaml file is created.

5. Install SSL certificate and key chains:
  - When using own certificate, create a directory for the certificates.  
**Attention!** The certificate file name and key name should match the example below:

```
mkdir -p certs
cp /somewhere/my-certificate-chain.crt certs/express.crt
cp /somewhere/my-unencrypted-key.key certs/express.key
```

The /somewhere/my-certificate-chain.crt and /somewhere/my-unencrypted-key.key constructs are individual for each specific case.

The certs/express.crt and certs/express.key constructs are mandatory.

Requirements for certificates are set out on page 34;

- when using a Let's Encrypt certificate, add the parameter le\_email: [admin@company-mail.ru](mailto:admin@company-mail.ru) to the settings.yaml file.

Checking the connection of certificates after installation is described on page 70.

6. Configure DLPS to allow security administrators to access message content (for setup options, see page 67).
7. Install cAdvisor (installation is performed from the /opt/express directory):

```
dpl cadvinstall
ps ax|grep cadvisor | grep -v grep
```

Command output:

```
17605 ? Ssl 44:20 /usr/bin/cadvisor -listen_ip 172.17.0.1 -port 9100
```

8. Install Prometheus node exporter from the /opt/express directory using the following command:

```
dpl nxinstall
ps ax|grep node_exporter | grep -v grep
```

Command output:

```
17802 ? Ssl 322:51 /usr/bin/node_exporter --web.listen-address=172.17.0.1:9200
```

Once the installation of the CTS server and supporting software is complete, a configuration file is created in which you need to set parameters for connecting to the RTS server, receiving push notifications, SMS messages and other functions.

The configuration file created by default looks like this and requires editing:

```
api_internal_token: verystrongpassword
ccs_host: cts_name.somedomain.sometld
cts_id: 'aaaa-bbbb-cccc-dddd'
phoenix_secret_key_base: verystrongpassword
postgres_password: verystrongpassword
prometheus_users:
  prometheus: verystrongpassword
rts_host: 'rts_name.somedomain.sometld'
rts_id: 'aaaa-bbbb-cccc-dddd'
rts_token: 'verystrongpassword'
```

**To change the configuration file**, use any text editor and make the following corrections to the file. The list of all settings in the configuration file is provided in [Table 35](#).

Table 35

Setting name	Value
<b>Mandatory settings</b>	
ccs_host	The full domain name of this server, which is registered in DNS and the corresponding name for which the certificate was purchased
cts_id	ID of the installed server, provided by the developer
prometheus_users	List of users with passwords generated by the htpasswd utility to access the Prometheus stack integrated into the system
rts_host	Full domain name of the RTS server to which the installed CTS server will connect (provided by the developer)
rts_id	ID of the RTS Server (provided by the developer)
rts_token	Token for authorization on the RTS server (provided by the developer)
le_email	This parameter is set when using a certificate from Let's Encrypt. The value of the parameter shall correspond to the e-mail to which notifications from Let's Encrypt will be sent
janus_enabled	Set to true
turnserver_shared_key	Key for connecting the Media server to the CTS server (generated during <a href="#">Media server installation</a> )
admin_url	This parameter is specified to override the standard path (/admin) to the administrator web interface: for example /not-admin
sip_trunk_enable: true	This parameter is set to use calls via SIP telephony. After adding the parameter, run the following command in the /opt/express directory: <pre>- dpl -d messaging ss -stuln   grep 5060</pre>
<b>Optional settings</b>	
Access to administrator web interface and DLPS	admin_allow: - 10.0.0.0/8

Setting name	Value
administrator console	- 172.16.1.0/24
Access to Prometheus	prometheus_allow: - 10.0.0.0/8 - 172.16.1.0/24
Changing the path to the administrator interface	admin_url: /express-admin-ui
Changing the path to the DLPS administrator interface	dlps_url: /dlps-admin-ui

The steps to connect the Media Server to the CTS server are described in the “[Connecting Media Server to CTS](#)” section.

**For the correct operation of the server**, it is recommended to correct the `cts_id`, `rts_host`, `rts_id` and `rts_token` parameters; in the example above they are highlighted in red.

**Note.** The values of the `cts_id`, `rts_host`, `rts_id` and `rts_token` parameters shall be put in quotation marks ('value'). This requirement does not apply to other parameters. To prevent errors, it is recommended to replace the parameters of the generated file with the parameters issued by the developers. When entering values manually, the quote characters are not entered.

## FRONT CTS AND BACK CTS INSTALLATION

The combination of Front CTS and Back CTS shall be installed in a certain order.

### To install Front CTS:

1. Open the Command Prompt.
2. Connect to the developer's Docker repository to download containers:

```
docker login -u Login -p Password registry.public.express
```

**Note.** Login and Password, which are issued by the developer, are used as login and password.

3. Download the container installer:

```
docker run --rm registry.public.express/dpl:cts-release dpl-install | bash
```

4. A YAML file with containers and an installer will be downloaded from the repository to the server.
5. Create a working directory for Front CTS:

```
mkdir -p /opt/express  
cd /opt/express  
echo DPL_IMAGE_TAG=cts-release > dpl.env  
dpl --init
```

6. Install SSL certificate and key chains:

- When using own certificate, create a directory for the certificates.

**Attention!** The certificate file name and key name should match the example below:

```
mkdir -p certs  
cp /somewhere/my-certificate-chain.crt certs/express.crt  
cp /somewhere/my-unencrypted-key.key certs/express.key
```

The `/somewhere/my-certificate-chain.crt` and `/somewhere/my-unencrypted-key.key` constructs are individual for each specific case.

The certs/express.crt and certs/express.key constructs are mandatory.

Requirements for certificates are set out on page 34;

- when using a Let's Encrypt certificate, add the parameter le\_email: [admin@company-mail.ru](mailto:admin@company-mail.ru) to the settings.yaml file.

Checking the connection of certificates after installation is described on page 70.

7. Open the settings.yaml configuration file for editing and add the following parameters:

```
api_internal_token: verystrongpassword
ccs_host: cts_name.somedomain.sometld
cts_id: 'aaaa-bbbb-cccc-dddd'
phoenix_secret_key_base: verystrongpassword
postgres_password: verystrongpassword
prometheus_users:
  prometheus: verystrongpassword
rts_host: 'rts_name.somedomain.sometld'
rts_id: 'aaaa-bbbb-cccc-dddd'
rts_token: 'verystrongpassword'
```

**For the correct operation of the server**, it is recommended to correct the cts\_id, rts\_host, rts\_id and rts\_token parameters; in the example above they are highlighted in red.

**Note.** The values of the cts\_id, rts\_host, rts\_id and rts\_token parameters shall be put in quotation marks ('value'). This requirement does not apply to other parameters. When entering values manually, the quote characters are not entered.

8. Edit the settings.yaml configuration file and add the following parameters:

```
cts_frontend: true
kafka_host: backend_name.somedomain.sometld
postgres_host: backend_name.somedomain.sometld
frontend_host: frontend_name.somedomain.sometld
backend_host: backend_name.somedomain.sometld
```

### To install Back CTS:

1. Open the Command Prompt.
2. Connect to the developer's Docker repository to download containers:

```
docker login -u Login -p Password registry.public.express
```

**Note.** Login and Password, which are issued by the developer, are used as login and password.

3. Download the container installer:

```
docker run --rm registry.public.express/dpl:cts-release dpl-
install | bash
```

A YAML file with containers and an installer will be downloaded from the repository to the server.

4. Create a working directory for Back CTS:

```
mkdir -p /opt/express
cd /opt/express
```

5. Install SSL certificate and key chains:

- When using own certificate, create a directory for the certificates.

**Attention!** The certificate file name and key name should match the example below:

```
mkdir -p certs
```



```
cp /somewhere/my-certificate-chain.crt certs/express.crt
cp /somewhere/my-unencrypted-key.key certs/express.key
```

The /somewhere/my-certificate-chain.crt and /somewhere/my-unencrypted-key.key constructs are individual for each specific case.

The certs/express.crt and certs/express.key constructs are mandatory.

Requirements for certificates are set out on page 34;

- when using a Let's Encrypt certificate, add the parameter le\_email: [admin@company-mail.ru](mailto:admin@company-mail.ru) to the settings.yaml file.

Checking the connection of certificates after installation is described on page 70.

6. Copy the configuration file from Front CTS (/opt/express/settings.yaml) to Back CTS and place it in the /opt/express folder.
7. Open the configuration file settings.yaml for editing (the file uses the YAML markup language):

```
api_internal_token: verystrongpassword
ccs_host: cts_name.somedomain.sometld
cts_id: 'aaaa-bbbb-cccc-dddd'
phoenix_secret_key_base: verystrongpassword
postgres_password: verystrongpassword
prometheus_users:
  prometheus: verystrongpassword
rts_host: 'rts_name.somedomain.sometld'
rts_id: 'aaaa-bbbb-cccc-dddd'
rts_token: 'verystrongpassword'
cts_frontend: true
kafka_host: backend_name.somedomain.sometld
postgres_host: backend_name.somedomain.sometld
frontend_host: frontend_name.somedomain.sometld
backend_host: backend_name.somedomain.sometld
```

**Note.** The values of the cts\_id, rts\_host, rts\_id and rts\_token parameters shall be put in quotation marks ('value'). This requirement does not apply to other parameters. When entering values manually, the quote characters are not entered.

The list of all settings in the configuration file is provided in Table 36.

Table 36

Setting name	Value
<b>Mandatory settings</b>	
ccs_host	The full domain name of this server, which is registered in DNS and the corresponding name for which the certificate was purchased
cts_id	ID of the installed server, provided by the developer
prometheus_users	List of users with passwords generated by the htpasswd utility to access the Prometheus stack integrated into the system
rts_host	Full domain name of the RTS server to which the installed CTS server will connect (provided by the developer)
rts_id	ID of the RTS Server (provided by the developer)
rts_token	Token for authorization on the RTS server (provided by the developer)
le_email	This parameter is set when using a certificate from Let's Encrypt. The value of the parameter shall correspond to the e-mail to which notifications from Let's Encrypt will be sent
janus_enabled	Set to true
turnserver_shared_key	Key for connecting the Media server to the CTS server (generated during <a href="#">Media server installation</a> )
admin_url	This parameter is specified to override the standard path (/admin) to the administrator web interface: for example /not-

Setting name	Value
	admin
sip_trunk_enable: true	This parameter is set to use calls via SIP telephony. After adding the parameter, run the following command in the /opt/express directory: <pre>- dpl -d messaging ss -stuln   grep 5060</pre>
<b>Optional settings</b>	
Access to administrator web interface and DLPS web interface	admin_allow: - 10.0.0.0/8 - 172.16.1.0/24
Access to Prometheus	prometheus_allow: - 10.0.0.0/8 - 172.16.1.0/24
Changing the path to the administrator interface	admin_url: /express-admin-ui
Changing the path to the DLPS administrator interface	dlps_url: /dlps-admin-ui

**To change the configuration file**, use any text editor and make the following corrections to the file.

- When editing the configuration file, remove additional settings:

```
cts_frontend: true
kafka_host: backend_name.somedomain.sometld
postgres_host: backend_name.somedomain.sometld
```

add the following parameter:

```
cts_backend: true
set_real_ip_from:
- ip_frontend
```

where ip\_frontend is the IP address of the Front server.

Edit the parameters by substituting the appropriate values:

- frontend\_name.somedomain.sometld;
- backend\_name.somedomain.sometld:

```
kafka_advertised_host_name: backend_name.somedomain.sometld
```

- Install Prometheus node exporter from the /opt/express directory using the following command:

```
dpl nxinstall
ps ax|grep node_exporter | grep -v grep
```

The steps to connect the Media Server to the CTS server are described in the ["Connecting Media Server to CTS"](#) section.

**Attention!** In the event that Internet access from Back CTS must be restricted for information security reasons, the use of TinyProxy is provided. If you need to use proxy, we recommend that you familiarize yourself with instructions for setting up proxy for the Docker service at the following link: <https://docs.docker.com/config/daemon/systemd/>.

### To install TinyProxy:

- In the directory where the OS is installed, run the following command:

```
Ubuntu\Debian - apt-get install -y tinyproxy
RHEL\CentOS - yum install -y epel-release
RHEL\CentOS - yum install -y tinyproxy
```

- Create a file /etc/tinyproxy/filter which lists the hosts to be accessed through the proxy:

```
registry.public.express
```

```
registry-auth.public.express
```

A configuration file for tinyproxy will be created automatically: /etc/tinyproxy/tinyproxy.conf.

3. Add the following settings to the /etc/tinyproxy/tinyproxy.conf file:

```
User tinyproxy
Group tinyproxy
Port 8888
Timeout 600
DefaultErrorFile "/usr/share/tinyproxy/default.html"
StatFile "/usr/share/tinyproxy/stats.html"
LogFile "/var/log/tinyproxy/tinyproxy.log"
LogLevel Info
PidFile "/var/run/tinyproxy/tinyproxy.pid"
MaxClients 300
MinSpareServers 5
MaxSpareServers 10
StartServers 3
MaxRequestsPerChild 0
#BackIP
Allow 192.168.80.22
ViaProxyName "tinyproxy"
Filter "/etc/tinyproxy/filter"
FilterDefaultDeny Yes
ConnectPort 443
ConnectPort 563
```

4. Restart the tinyproxy service using the following command:

```
systemctl restart tinyproxy
```

## CONNECTING MEDIA SERVER TO CTS

### To set up Media server connection to CTS:

1. Connect to the CTS (Single/Back) server via SSH.
2. Specify the turnserver\_shared\_key value in /opt/express/settings.yaml (the key is generated during [Media server installation](#), YmNjY2VmNDk0ZTEwNDgzNj is used as an example):

```
turnserver_shared_key: YmNjY2VmNDk0ZTEwNDgzNj
```

3. Remove the configuration files of the group calling service (janus) by running the following commands:

```
cd /opt/express-voice && dpl --dc down
cd ~ && rm -rf /opt/express-voice
```

## SETTING UP MEDIA SERVER

The procedure for setting up Media server includes:

- [setting up JANS, STUN and TURN servers](#) (mandatory setting);
- [setting up IP telephony](#) (optional).

## SETTING UP JANS, STUN AND TURN SERVERS

### To set up JANS, STUN and TURN Servers:

1. Go to the following directory: cd /opt/express-voice/:

```
cd /opt/express-voice
```

2. Start the Media server via the command line with the following command:

```
dp1 -d
```

3. Open the administrator web interface.
4. In the VoEx section, to enable the audio-video calling feature in the "Janus instances" section (see [Figure 10](#)), add Media server names in the format `ws://internal_fqdn_media_cts:8188` for each server separately. In the "Janus External Host" section, enter the public IP of the Media server.

Figure 10

5. Disable the old Janus server settings.
6. in the "TURN Server (comma separated)" field, enter the external FQDN of your Server and the port number separated by a colon, for example: `,express.firma.ru:3478`;
7. in the "STUN Server (comma separated)" field, enter the external FQDN of your Server and the port number separated by a colon, for example: `,express.firma.ru:3478`.
8. In the "Local VoEX network" field, specify the local network mask (see [Figure 11](#)).

Figure 11

9. Check the following boxes if necessary. A description of the settings is provided in [Table 37](#):

Table 37

Setting up	Description
Allow screen sharing outside of the closed contour	Allows the users to share their device screens with other users outside the CTN (RTS server users, trust server users, users who have left the CTN zone)
Use only Relay Ice candidates	Forced use of TURN server
Allow TCP ICE	The mark is set — TCP connection in TURN server is allowed. The mark is not set — TCP connection in TURN server is not allowed.
Enable audio stream mixing	Combines audio streams of calls directed from users to the server into one stream
Enable VP9 video codec	Item under development
Enable use of internal janus host for servers	Use the internal Janus host for the servers specified in the field below (see item 10)
Enable the ability to record calls	Allows users to record individual and group calls

---

**Note.** It is recommended to check “Allow screen sharing outside of the closed contour” and “Enable mixing of audio streams”.

---

10. In the “List of servers that will use the internal Janus host (comma separated)” field, enter a list of the CTS IDs of the servers with which communication will be conducted via the internal host.
11. Select the recording mode from the drop-down list.
12. Click “Save”.

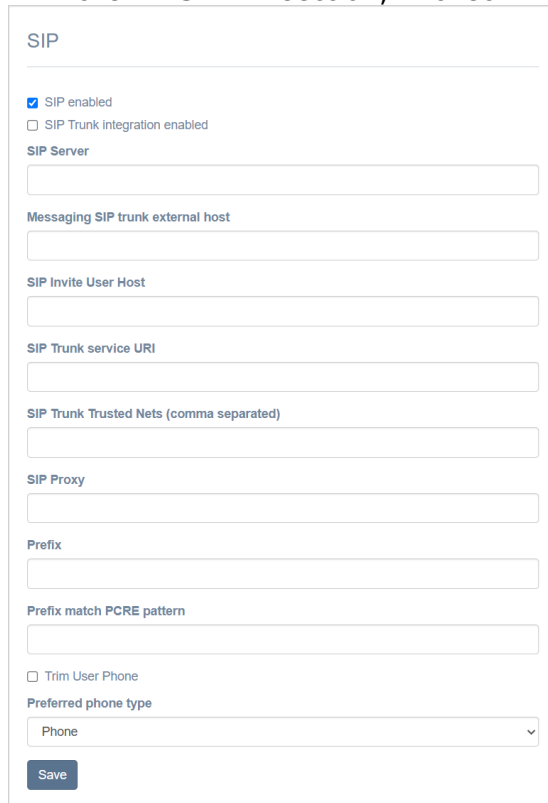
**To start the Media server**, run the commands, which are similar to the commands for starting the CTS server set out on page 70. The Media server installation commands shall be run from the /opt/express-voice/ directory.

---

## SETTING UP IP TELEPHONY

**To set up IP telephony:**

1. In the "SIP" section, check the "SIP enabled" box (see



SIP

☒ SIP enabled  
☐ SIP Trunk integration enabled

SIP Server

Messaging SIP trunk external host

SIP Invite User Host

SIP Trunk service URI

SIP Trunk Trusted Nets (comma separated)

SIP Proxy

Prefix

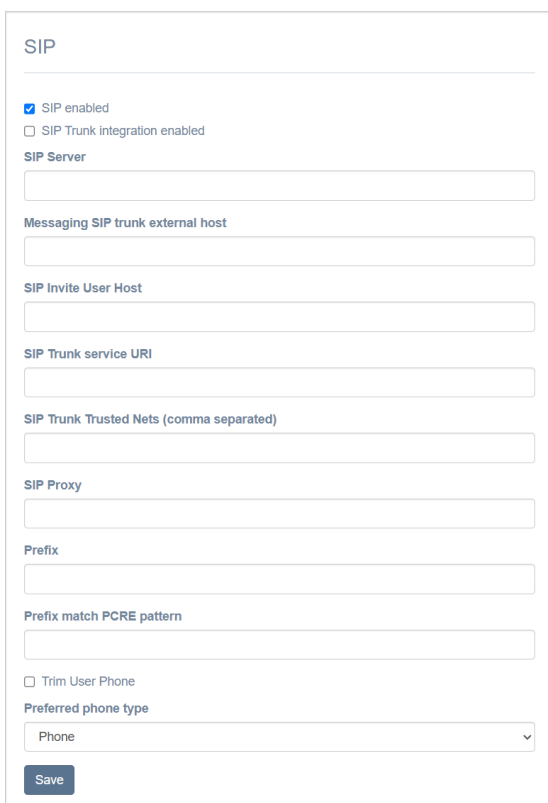
Prefix match PCRE pattern

☐ Trim User Phone

Preferred phone type

Save

2. [Figure 12](#)).



SIP

☒ SIP enabled  
☐ SIP Trunk integration enabled

SIP Server

Messaging SIP trunk external host

SIP Invite User Host

SIP Trunk service URI

SIP Trunk Trusted Nets (comma separated)

SIP Proxy

Prefix

Prefix match PCRE pattern

☐ Trim User Phone

Preferred phone type

Save

[Figure 12](#)

3. Fill in the fields. A description of the fields is provided in [Table 38](#):

[Table 38](#)

Field	Purpose
SIP server	ATE domain name or IP address (SIP trunk). If the port is other than UDP/5060, specify it separated by a colon
Host, which is added to username when registering the SIP terminal	A field that is transmitted in the invite message towards the ATE. By default, the ccs_host value is added. If necessary, specify the host address from the configuration file
URI to connect to SIP Trunk	Back CTS address where the messaging container is installed. Filled out for Media and Back CTS deployments. Record format: sip:<IP or DNS name>:<port>
List of allowed SIP Trunk addresses	IP addresses from which calls will be received by the eXpress CS IP trunk. Specify at least two IP addresses: <ul style="list-style-type: none"> <li>ATE IP address;</li> <li>address where the janus container is installed (SIP gateway, which is installed alongside with eXpress CS).</li> </ul> All IPs or networks shall be indicated with a mask, for example – 10.10.10.1/32 for a single IP, 192.168.12.0/24 for a network. To deploy Single CTS, specify the IP address of the eXpress CS server itself (10.10.10.1/32) and the internal IP of the docker network interface (172.18.0.1/32) and the ATE. For Media and Back CTS deployment, specify IP Media and PBX
SIP Proxy	SIP telephony proxy server address or ATE address. SIP record format: <IP or DNS name>:<port>. It is not necessary to specify the port if it does not differ from standard UDP/5060
Prefix	A string inserted at the beginning of the dialed number when transmitting the number to the ATE and the number received from the ATE if the ATE sends the number without a prefix
PCRE template for prefix substitution	A regular expression to match the structure of the number to which a prefix will be inserted when making an outgoing call from eXpress CS. To prevent the prefix from being added to numbers, enter the expression <code>^[0-9](1)</code>
Preferred phone type	The type of phone from which calls will be made. Possible options: <ul style="list-style-type: none"> <li>phone;</li> <li>IP phone;</li> <li>phone (other);</li> <li>IP-phone (other).</li> </ul> The mapping of user object settings to these phone types is configured in the Active Directory section of the administrator web interface. The selected phone type will be hidden in the server user profiles when SIP integration is disabled.

4. Click "Save".

Next, configure SIP trunk of the client ATE.

---

**Attention!** For all deployment schemes, it is mandatory to disable SIP trunk status checking.

---

**For correct operation with the Single CTS <sup>1</sup> deployment scheme, configure ATE 2 SIP trunk:**

1. For calls from ATE to the System, specify the Single CTS destination IP.
2. For calls from the System to the ATE, specify the Media destination IP.

---

<sup>1</sup>ATE network interactions diagram for Single CTS deployment scheme and network interactions for this deployment scheme are provided in [Appendix 8](#).

### For correct operation with Front CTS and Back CTS <sup>1</sup> deployment scheme, configure ATE 2 SIP trunk:

1. For calls from ATE to the System, specify the Back CTS destination IP.
2. For calls from the System to the ATE, specify the Media destination IP.

## LINKS SERVER INSTALLATION

### To install a Links Server:

1. Open the Command Prompt.
2. Connect to the developer's Docker repository to download containers:

```
docker login -u Login -p Password registry.public.express
```

**Note.** Login and Password, which are issued by the developer, are used as login and password.

3. Create a working directory for Web Client:

```
mkdir -p /opt/xlnk  
cd /opt/xlnk  
echo DPL_IMAGE_TAG=xlnk-release > dpl.env  
dpl --init
```

4. After running the dpl --init command, a settings.yaml file is created.
5. Install SSL certificate and key chains:

- When using own certificate, create a directory for the certificates:

**Important!** The certificate file name and key name should match the example below:

```
mkdir -p certs  
cp /somewhere/my-certificate-chain.crt certs/express.crt  
cp /somewhere/my-unencrypted-key.key certs/express.key
```

The /somewhere/my-certificate-chain.crt and /somewhere/my-unencrypted-key.key constructs are individual for each specific case.

The certs/express.crt and certs/express.key constructs are mandatory.

Requirements for certificates are set out on page [34](#);

- when using a Let's Encrypt certificate, add the parameter le\_email: [admin@company-mail.ru](mailto:admin@company-mail.ru) to the settings.yaml file.

Checking the connection of certificates after installation is described on page [70](#).

The configuration file created by default looks like this and requires editing:

```
ccs_host: somehost.somedomain.sometld
```

Example of configuration settings:

```
ccs_host: xlnk.example.com  
le_email: test@example.com  
home_address: www.example.com
```

<sup>1</sup>ATE network interactions diagram with Front CTS + Media and Back CTS and network interactions for this deployment scheme are provided in [Appendix 9](#).



```
android_app_link:
'https://play.google.com/store/apps/details?id=ru.unlimitedtech.ex
press'
ios_app_link: 'https://apps.apple.com/ru/app/express-enterprise-
messaging/id1225251588?l=en'ets_id: 00000000-0000-000-000-
000000000000
api_gw_url: 'http://link:4000'
web_host_default: 'web.example.com'
```

The list of settings with descriptions is provided in [Table 39](#).

6. In the /opt/express/xlnk directory, run the following command

```
dpl -d
```

**To change the configuration file**, use any text editor and make the following corrections to the file (see [Table 39](#)):

*Table 39*

Setting name	Value
ccs_host	The full domain name of this server, which is registered in DNS and the corresponding name for which the certificate was purchased
le_email	This parameter is set when using a certificate from Let's Encrypt. The value of the parameter shall correspond to the e-mail to which notifications from Let's Encrypt will be sent
home_address	The full domain name of the company's main website, to which users will be redirected when contacting without a link to the chat/conference
ets_id	ETS server identifier, which is required to identify links created on enterprise servers. Enables the display of links to the company's mobile apps
android_app_link ios_app_link	Links to mobile apps in the Apple, Play Market app stores
android_app_name ios_app_name	The name of the link to mobile apps, by default the value is Android Custom App, iOS Custom App. Displayed when clicking on a link from mobile devices
api_gw_url	Path to xlink service for access
web_host_default	Full domain name of the web client server for chat/conference

## INSTALLING DLPS

### INSTALLING DLPS ON A DEDICATED SERVER

#### To generate a DLPS key and add it to all chats:

**Attention!** In this example, DLPS is installed on a server, which is separate from the CTS server.

On the CTS server, specify the external DLPS.

To install Docker, on the Back server, in the file /opt/express/settings.yaml add the following:

```
dlps_external: true
dlps_host: fqdn_dlps
```

After making changes on the same server, run the following command:

```
cd /opt/express/ && DPL_PULL_POLICY=never dpl -p && DPL_PULL_POLICY=never dpl
--dc exec nginx nginx -s reload
```

To install in k8s, fill in the following parameter in the chart in values.yaml:

```
dlps_host: fqdn_dlps
```

1. Open the Command Prompt.
2. Connect to the developer's Docker repository to download containers:

```
docker login -u Login -p Password registry.public.express
```

**Note.** Login and Password, which are issued by the developer, are used as login and password.

3. Download the container installer:

```
docker run --rm registry.public.express/dpl:dlps-release dpl-  
install | bash
```

4. Create a working directory for Web Client:

```
mkdir -p /opt/express  
cd /opt/express  
echo DPL_IMAGE_TAG=dlps -release > dpl.env  
dpl --init
```

After running the dpl --init command, a settings.yaml file is created.

5. Install SSL certificate and key chains.

- When using own certificate, create a directory for the certificates.

**Important!** The certificate file name and key name should match the example below:

```
mkdir -p certs  
cp /somewhere/my-certificate-chain.crt certs/express.crt  
cp /somewhere/my-unencrypted-key.key certs/express.key
```

The /somewhere/my-certificate-chain.crt and /somewhere/my-unencrypted-key.key constructs are individual for each specific case.

The certs/express.crt and certs/express.key constructs are mandatory.

Requirements for certificates are set out on page [34](#).

- when using a Let's Encrypt certificate, add the parameter le\_email: [admin@company-mail.ru](mailto:admin@company-mail.ru) to the settings.yaml file.

Checking the connection of certificates after installation is described on page [70](#).

6. Run the following command:

```
mkdir -p dlps_keys/storage && chown -R 888:888 dlps_keys
```

The configuration file created by default looks like this and requires editing:

```
api_internal_token: token  
ccs_host: somehost.somedomain.sometld  
cts_id: ''  
dlps_host: ''  
dlps_icap_client_host: ''  
dlps_icap_additional_headers: {}  
etcd_endpoints: http://etcd:2379  
kafka_host: kafka  
phoenix_secret_key_base: token  
postgres_endpoints: ''  
postgres_user: ''  
postgres_password: ''  
redis_connection_string: ''  
rts_id: ''dlps_enabled: true
```

The list of settings with descriptions is provided in [Table 40](#).

7. Run the following command (from the /opt/express folder):

```
dpl -d
```

After running this command, a key will be generated that will be added to all chats.

**To change the configuration file**, use any text editor and make the following corrections to the file.

Table 40

Setting name	Value
<b>Mandatory settings</b>	
ccs_host	The full domain name of the CTS server, which is registered in DNS and the corresponding name for which the certificate was purchased
le_email	This parameter is set when using a certificate from Let's Encrypt. The value of the parameter shall correspond to the e-mail to which notifications from Let's Encrypt will be sent
cts_id	Installed server ID
rts_id	ID of the RTS Server (provided by the developer)
etcd_endpoints	ETCD server connection address
kafka_host	Kafka server connection address
redis_connection_string	REDIS database connection parameters
postgres_endpoints postgres_user postgres_password	PostgreSQL database connection parameters
dlps_postgres_endpoints dlps_postgres_user dlps_postgres_password	In case of using a separate database for the DLPS module, it shall be additionally specified
<b>Optional settings</b>	
Access to the DLPS web interface	admin_allow: - 10.0.0.0/8 - 172.16.1.0/24
Changing the path to the DLPS administrator interface	dlps_url: /dlps-admin-ui

## INSTALLING DLPS ON SINGLE CTS

### To generate a DLPS key and add it to all chats:

**Attention!** In this example, DLPS is installed on Single CTS. For installation diagrams for other DLPS locations, please contact the developers.

- Run the following command:  

```
mkdir -p dlps_keys/storage && chown -R 888:888 dlps_keys
```
- Specify the parameter "dlps\_enabled: true" in the configuration file:  

```
api_internal_token: S0L2U6zD0s2iQmdQ
ccs_host: cts_name.somedomain.sometld
cts_id: 'aaaa-bbbb-cccc-dddd'
phoenix_secret_key_base: verystrongpassword
prometheus_users: verystrongpassword
prometheus: verystrongpassword
rts_host: 'rts_name.somedomain.sometld'
rts_id: 'aaaa-bbbb-cccc-dddd'
rts_token: 'verystrongpassword'
dlps_enabled: true
```
- Run the following command (from the /opt/express folder):  

```
dpl -d && dpl --dc restart nginx
```
- After running this command, a key will be generated that will be added to all chats.
- The administrator console will be accessible via the following URL: <https://express.mydomain.tld/dlps/>. The standard account is admin/admin.

6. In the administrator console, enable the DLPS setting by clicking on the "Enable DLPS" button (see [Figure 13](#)).

Settings

☒ DLPS enabled  
☐ Send stealth events

Audit log cleaning schedule (cron format)

Clear events older than the specified value in days. 0 clears all events

Administrator idle time before session termination (in seconds)

1800

Save

Figure 13

## INSTALLING DLPS ON SINGLE CTS WITH KEYS STORED ON EXTERNAL MEDIA

### To set up DLPS on external media:

1. Insert a rewritable (RW) USB flash drive into the computer and mount the drive to the desired directory. The default directory is `/opt/express-dlps/dlps_keys/`. The file system on the flash drive shall be compatible with RHEL OS.
2. Write the `dlps_keys_mount_path` setting in the configuration file as follows: `/PATH_TO_DIRECTORY`, where "PATH\_TO\_DIRECTORY" is the path to the directory where the keys are written.

For example:

```
api_internal_token: TOKEN
ccs_host: cts_name.somedomain.sometld
cts_id: 'aaaa-bbbb-cccc-dddd'
dlps_icap_client_host: IP_ADDRESS
dlps_icap_client_port: PORT
dlps_icap_additional_headers: verystrongpassword
network_segment: CTS
application: PROD
client_ip: 127.0.0.1
server_ip: 127.0.0.1
kafka_host: etcd01.ru,etcd02.ru,etcd03.ru
phoenix_secret_key_base: PHOENIX_SECRET_KEY_BASE
etcd_endpoints:
http://etcd01.ru:2379,http://etcd02.ru:2379,http://etcd03.ru:2379
postgres_host: CTS.CTS.RU
postgres_user: POSTGRES_USER
postgres_password: POSTGRES_PASSWORD
dlps_keys_mount_path: /MOUNT_POINT
prometheus_users: verystrongpassword
prometheus: verystrongpassword
rts_id: 'aaaa-bbbb-cccc-dddd'
pacemaker_generate: true
pacemaker_virtual_ip: 10.0.0.1
```

3. Run the following command:

```
mkdir -p dlps_keys/storage && chown -R 888:888 dlps_keys
```

4. Start DLPS (if DLPS is already running, stop and restart it):

```
dpl -d
```

- To verify the correctness of installation, make sure that the correct "volumes" value is specified in the /opt/express-dlps/dlps/docker-compose.yml file: "/PATH\_TO\_DIRECTORY:/app/keys".

## INSTALLING CALL AND CONFERENCE RECORDING COMPONENTS

### Note.

- Before installation, you need to update the CTS server to version 3.10 or higher<sup>1</sup>;
- Before installing components, it is recommended to familiarize yourself with the [architecture](#);
- It is necessary to open network access from the Media server to CTS Back server via port 443.

### To install components:

- On Back CTS or Single CTS, add the following to /opt/express/settings.yaml:

```
transcoding_enabled: true
```

- On Back CTS or Single CTS, run the following command:

```
cd /opt/express/ && dpl -p && dpl -d transcoding_manager  
recordings_bot admin && dpl --dc restart nginx
```

- On the Media server, add the following to the /opt/express-voice/settings.yaml file:

**Note.** Copy the value ccs\_host, api\_internal\_token from /opt/express/settings.yaml located on Back CTS or Single CTS.

```
transcoding_hosts:  
  cts:  
    ccs_host: cts.corp.express  
    api_internal_token: token-cts
```

If the recording server and janus are used by multiple CTS servers, list multiple hosts:

```
transcoding_hosts:  
  cts1:  
    ccs_host:  
      cts1.corp.express  
    api_internal_token: token-cts1  
  cts2:  
    ccs_host: cts2.corp.express  
    api_internal_token: token-cts2
```

- On the Media server, run the following command:

```
cd /opt/express-voice/ && dpl -p && dpl -d
```

<sup>1</sup> Setting up call and conference recording is described in the eponymous section of the document "Administrator's Guide. Volume 2. Operation of the CTS Server".

## CERTIFICATE VERIFICATION

**To test the correctness of the certificate** after installing the product, run the following command:

```
openssl s_client -connect fqnd-cts:443
```

The following message indicates an error:

```
depth=0 CN = *.domain.ru
verify error:num=20:unable to get local issuer certificate
verify return:1
depth=0 CN = *.domain.ru
verify error:num=21:unable to verify the first certificate
verify return:1
```

## STARTING UP THE SERVER

### To start up the server:

**Note.** Commands to start the server shall be run from the installation directory /opt/express.

1. Run the following command:

**Note.** In case of split installation, this command shall be run first on the Back CTS server, then on the Front CTS server.

```
dpl -d
```

2. Check if all containers are running using the following command:

```
docker ps -a
```

If the containers are not running, run the following command to view the event log:

```
dpl --dc logs --tail=200 <container_that_was_not_launched>
```

As long as the server installation procedure is performed correctly, the administrator web interface will be installed and available within five minutes: [https://ccs\\_host/admin](https://ccs_host/admin).

**Note.** For correct operation of the administrator web interface, it is **not recommended** to use Internet Explorer.

3. Create an administrator account. The command shall be issued on Back CTS:

```
dpl --dc exec admin bin/admin add_admin -u admin -p
'veryinsecurepassword123'
```

**Note.** Requirements for administrator password:

- the minimum password length is 8 characters;
- The password must contain at least one special character #!?\$%^&\*(), one lowercase and one uppercase letter.

If the administrator web interface does not install, a password policy mismatch error has occurred. In this case, as well as in case of other errors, it is necessary to perform a check.

**To check for errors**, in the logs that appear, it is necessary to find the most frequent mention with errors and restart the container generating the error with the following command:

```
dpl --dc restart {container_name}
```

For example:

```
dpl --dc restart nginx
```

**Note.** All container names corresponding to a specific architecture are listed in the [Architecture](#) section.

If this operation does not help, contact the technical support of the developer company.

## Chapter 3

### SETTING UP THE SERVER

For normal operation of the system, it is necessary to pre-configure the server in the administrator web interface. The setup procedure depends on the server type and is described in the relevant sections below:

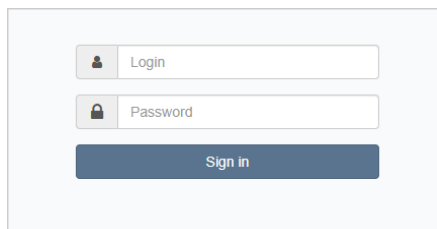
- [ETS](#);
- [CTS](#).

#### To authorize in the administrator web interface:

1. In the address bar of your browser, enter the address of the administrator web interface.

**Important!** For ETS, login is performed in the administrator web interface, see [https://ets\\_host/admin](https://ets_host/admin), for CTS — see [https://cts\\_host/admin](https://cts_host/admin). Without https the administrator web interface is not accessible.

An authorization window will open (see [Figure 14](#)):



*Figure 14*

2. Enter the account name and password in the appropriate fields.
3. Click "Login".

The main window of the administrator web interface will open.

**To exit the administrator web interface**, click  in the upper left part of the window.

### SETTING UP THE ETS SERVER

The procedure for setting up the ETS server includes the following:

- [Connecting the TLS Certificate](#) (if this has not been done during the ETS installation process);
- [setting up video and voice communication](#);
- [Connecting the SMTP Server](#);
- [setting up push notifications](#);
- [setting up SMS service](#);
- [Connecting ETS Administrators from AD](#);
- [setting up CTS connections](#).



## CONNECTING THE TLS CERTIFICATE

**To set up the TLS certificate**, in the administrator web interface, select the "Server" menu item. A window will open with information about this ETS server (see [Figure 15](#)).

The screenshot displays the ETS server administrator web interface. The 'Trusts TLS Certificate' section is active, showing fields for 'Certificate' and 'Key', both with 'Choose File' buttons and 'No file chosen' text. A 'Save' button is present. To the left, the 'Server Settings' section includes fields for 'Avatar', 'Mobile background', 'Mobile dark background', 'Web background', 'Web dark background', 'Web high resolution background', and 'Web dark high resolution background', each with a 'Choose File' button and 'No file chosen' text. A 'Hide server name' checkbox and a 'Save' button are also visible. To the right, the 'Service versions' section lists various services and their versions: admin 3.36.0, authentication 3.36.0, email\_notifications 3.36.0, kdc 3.36.0, messaging 3.36.0, phonebook 3.36.0, file\_service 3.36.3, push\_service 3.36.0, settings 3.36.0, sms\_service 3.36.0, trusts 3.36.1, voex 3.36.0, and metrics\_service 3.36.0. The 'Admin Info' section at the bottom includes fields for 'Full name', 'Phone', 'Address', and 'Emails (comma separated)', with a 'Save' button.

Figure 15

### To use the TLS protocol in trusted connections:

1. Enter the information about the certificate and the key in the appropriate fields in the Trust TLS Certificate area.
2. Click "Save".

**Note.** It is allowed to use the TLS certificate used during the CTS server installation stage.

## SETTING UP VIDEO AND VOICE COMMUNICATION

Setting up video and voice communication is performed after installing the Media server and is described on page [59](#).

## CONNECTING THE SMTP SERVER

### To connect the SMTP Server:

1. Select "E-mail" from the menu.  
The "E-mail Settings" window will open for entering parameters (see [Figure 16](#)).

Figure 16

- In the “E-mail Settings” area, fill in the fields. A description of the fields is provided in [Table 41](#):

Table 41

Field	Description
App name	The name of the app from which e-mails will be sent
From	Return address
Server	FQDN or IP address of the mail server
Port	Port number for retransmission of outgoing mail: 25, 587 or 465. The port number depends on the type of connection
User name	E-mail address
Password	Data for authorization on the SMTP server. If authentication on the mail server is not used, then leave these fields blank
Password confirmation	Data for authorization on the SMTP server. If authentication on the mail server is not used, then leave these fields blank
Connection protection	Type of secure connection (drop-down list: SSL, Start/TLS or empty value)
Send e-mail via	Drop-down list for selecting a server from which e-mails will be sent (if you select “Local settings” in the drop-down list, e-mails will be sent via the server configured in this window; if you select “RTS”, e-mails will be sent via RTS).

- Click “Save”.

**To check connection settings**, use the “Test E-mail Sending” area. Enter the recipient's address in the empty field and click “Send”.

## SETTING UP PUSH NOTIFICATIONS

**To connect and configure push notifications**, go to the “Push Service” section.

The interface is designed to connect push notifications (see [Figure 17](#)).

Push Platforms			
<a href="#">Create for Android RuStore</a> <a href="#">Create for Android HMS</a> <a href="#">Create for Android</a> <a href="#">Create for iOS</a> <a href="#">Create for Web</a>			
Platform ^ v	Package ID ^ v	Updated at ^ v	Expires at ^ v
ios_apns	ios-apns-12345	2024-11-22 10:39:29	
android_silent	android-silent-12345	2024-11-22 10:39:10	

Figure 17

A description of the interface is provided in [Table 42](#):

Table 42

Column name	Information
Platform	The platform on which push notifications are enabled
Package ID	eXpress app build package name
Update date	Date when push notifications settings were last changed
Expiration Date	Push notification expiration date

**To edit a connection**, click  and make changes in the window that opens.

**To delete the connection**, click .

The mechanism for enabling push notifications varies depending on the platform. Push notifications are connected as follows:

- for Android – via FCM;
- for Huawei – via Push Kit;
- for iOS – via APNS;
- for web apps – via FCM.

**Note.** For correct operation, access to APN Push services is required:

- Apple APN – [api.push.apple.com](https://api.push.apple.com);
- Google FCM – [fcm.googleapis.com](https://fcm.googleapis.com); [www.googleapis.com](https://www.googleapis.com);
- Huawei HMS – [push-api.cloud.huawei.com](https://push-api.cloud.huawei.com), [oauth-login.cloud.huawei.com](https://oauth-login.cloud.huawei.com);
- RuStore – [vkpns.rustore.ru](https://vkpns.rustore.ru).

When interacting with external systems (Huawei HCM, Apple APN, Google FCM), a push notification may contain the following data (see [Table 43](#)):

Table 43

Name	Information
group_chat_id	ID of the chat where the event occurred
chat_type	Chat type (chat(group_chat botx))
push_opts	Additional options (silent – processing should not cause a notification to appear, dnd – the notification should be displayed even if the chat is set to silent, for example, if the current user was mentioned in the message)
sync_id	Message ID in chat
event_type	Message type (message_new bot_command app_event)
event_version	Message version (set to 1 by default)
server_id	Server ID of the message sender (except Android)
sender	User ID of the message sender
push_tag_id	Tag ID (except Android)
cleaned_at	Message deletion date, filled in when the message is deleted
unread_messages_count	Unread message counter displayed on the app icon (except Android)
missed_calls_count	Missed call counter (except Android)
parent_group_chat_id	ID of the parent chat where the event occurred (UID) (for calls)
inserted_at	Call start time (for calls)

Name	Information
body	Notification text (for SmartApp) (if empty, then "New event in SmartApp")
meta	Notification metadata (for SmartApp)
name	Conference title
startAt	Conference start time

### To create a connection in Android:

1. Open the Firebase console.
2. In the project (menu "Project Overview"), where the keys for Android are configured, select "Project settings".
3. In the eXpress administrator web interface, in the "Push Service" section, click "Create for Android" in the upper right corner.

A window for creating a connection for the Android platform will open (see Figure 18).

Create push platform for android Back to list

Platform

Package ID

Connection pool size

☐ Proxy to the RTS

Expires at  
2024-11-18 12:00:00

FCM URL

FCM service\_account.json

FCM API Key

Save

Figure 18

4. Fill in the fields of the form as described in Table 44:

Table 44

Parameter	Description	Value
Platform	The platform on which push notifications are enabled	android_silent
Package ID	eXpress app build package name	
Maximum number of platform connections	Push platform connection pool size	If you leave the field blank, the default pool size will be 10.
Expiration Date	Push notification expiration date	
FCM URL	Firebase Cloud Messaging Server Address	<a href="https://fcm.googleapis.com/v1/projects/{fcmProjectID}/messages:send">https://fcm.googleapis.com/v1/projects/{fcmProjectID}/messages:send</a> The ProjectID value is taken from the Firebase console (Project Settings → General)
FCM	Service account JSON file	The file can be downloaded from the

Parameter	Description	Value
service_account.json		Firebase console (Project Settings → Service Account)
FCM API Key	The key is not provided or required on the latest version of the Firebase Cloud Messaging API (HTTP v1)	

- Click "Save".

### To create a connection on HMS Android:

- Click "Create for HMS Android".

A window for creating a connection for the Huawei platform will open (see [Figure 19](#)).

Create push platform for android\_hms Back to list

Platform

Package ID

Connection pool size

☐ Proxy to the RTS

App ID

Client secret

Save

Figure 19

- Fill in the fields of the form (see [Table 45](#)):

Table 45

Parameter	Description	Value
Platform	The platform on which push notifications are enabled	android_hms
Package ID	eXpress app build package name	
Maximum number of platform connections	Push platform connection pool size	If you leave the field blank, the default pool size will be 10.
App ID	App ID in the Push Kit console	
Client secret key	Key in the Push Kit console	

- Click "Save".

### To create a connection in iOS:

- Click "Create for iOS" in the upper right corner.

A window for creating a connection for the iOS platform will open (see [Figure 20](#)).

Create push platform for ios

Back to list

Platform

Package ID

Connection pool size

☐ Proxy to the RTS

Expires at (automatically filled from cert)

2024-11-18 12:00:00

Mode

Key

Cert

Topic

Save

Figure 20

- Fill in the fields of the form as described in Table 46:

Table 46

Parameter	Description	Value
Platform	The platform on which push notifications are enabled	<ul style="list-style-type: none"> <li>ios_apns (for alert push with apns certificate);</li> <li>ios_voex (for push notifications calls with VoIP certificate)</li> </ul>
Package ID	eXpress app build package name	
Maximum number of platform connections	Push platform connection pool size	If you leave the field blank, the default pool size will be set to 10.
Expiration Date	Push notification expiration date	
Mode	Push notification operating mode. Possible values for prod/dev	<ul style="list-style-type: none"> <li>dev (for beta build);</li> <li>prod (for release/prerelease)</li> </ul>
Key	Private key	
Cert	Certificate	
Topic	eXpress app build name	Package ID (for ios_apns); empty value (for ios_voex)

- Click "Save".

### To create a connection in the Web app:

- Open the Firebase console.
- In the Firebase console, create a project for the Web app.
- In the window that opens, click "Generate key pair".
- In the administrator web interface, in the "Push Service" section, click "Create for Web" in the upper right corner.

A window for creating a connection for the Web app will open (see [Figure 21](#)).

Figure 21

5. Fill in the fields of the form (see [Table 47](#)):

**Note.** In the “Platform” field, enter the value “web”.

Table 47

Parameter	Description	Value
Platform	The platform on which push notifications are enabled	<ul style="list-style-type: none"> <li>web;</li> <li>web_chrome;</li> <li>web_firefox;</li> <li>web_edge</li> </ul>
Package ID	eXpress app build package name	
Maximum number of platform connections	Push platform connection pool size	If you leave the field blank, the default pool size will be 10.
Expiration Date	Push notification expiration date	
FCM API Key	API key issued in the Firebase administrator console	
Public VAPID key	Public API key generated in the Firebase administrator console	
Private VAPID key	Private API key generated in the Firebase administrator console	
VAPID subject (URI or e-mail)	User E-mail address in Firebase	mailto:<e-mail of the Firebase account>

6. Click “Save”.
7. Repeat steps 1 to -6 for Chrome, and specify the “web\_chrome” value in the “Platform” field.  
Two entries will appear in the “Push Service” section (for two browsers).
8. In the Web app Docker image configuration file (WEB\_CLIENT\_CONFIG), change the gcmSenderId parameter to the value from Firebase.

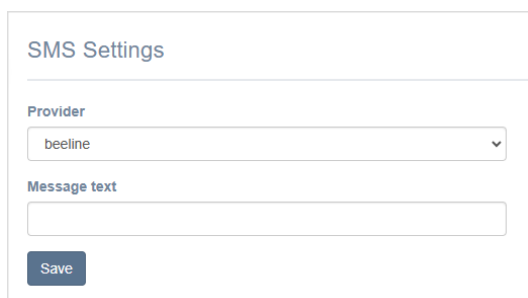
## SETTING UP SMS SERVICE

In the "SMS" section, the administrator can configure [the text of the message to be sent](#), [integration with the provider](#), which will send users SMS messages with an authorization code, and [security settings](#).

### SETTING UP THE TEXT OF SMS MESSAGES

#### To set up the text of SMS messages:

1. Select the "SMS" section in the menu.  
The "SMS Settings" window will open.
2. In the "Provider" field, select a provider. For example, Beeline.
3. In the "Text of SMS message" field, enter the text that will be sent along with the authorization code, and click "Save" (see [Figure 22](#)).



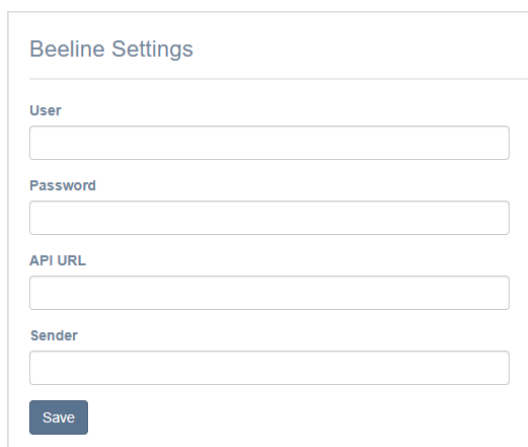
The screenshot shows a window titled "SMS Settings". Inside, there is a "Provider" dropdown menu with "beeline" selected. Below it is a "Message text" input field. At the bottom left is a "Save" button.

Figure 22.

### SETTING UP INTEGRATION WITH PROVIDER

#### Setting up integration with provider:

1. Go to the "Adapters" subsection.
2. Configure the parameters of the selected provider in the appropriate section, and click "Save" (see [Figure 23](#)).



The screenshot shows a window titled "Beeline Settings". It contains four input fields: "User", "Password", "API URL", and "Sender". A "Save" button is located at the bottom left.

Figure 23

The settings you can configure vary by provider. The examples of settings for providers are provided below in [Table 48](#):

Table 48

Parameter	Purpose	Provider
API key	Key for sending SMS messages.	Clickatell



Parameter	Purpose	Provider
	Provided by provider	
API URL	SMS service API address	Clickatell, QTelecom, Beeline, SMSTraffic
User	Username of the provider's SMS service	QTelecom, Beeline, SMSTraffic, Stream Telecom
Login	User login of the provider's SMS service	SMSC, Tele2
Password	Provider's SMS service user password	QTelecom, Beeline, SMSC, Tele2, SMSTraffic, Stream Telecom
Sender	SMS sender name (e.g. eXpress)	QTelecom, Beeline, SMSC, SMSTraffic
Sender for MTS	SMS sender name (e.g. eXpress)	QTelecom
Shortcode	Provided by provider	Tele2
SID	Provided by provider	Twilio
Token	Provided by provider	Twilio
From	SMS sender name	Stream Telecom
Validity	Message validity period	Stream Telecom
Callback URL	Address of the script to which POST data about the SMS delivery status is returned	Stream Telecom
User	Digital client ID that is returned to the address specified in the Callback_url parameter	Stream Telecom
Name deliver	Mailing name assigned for ease of searching in statistics	Stream Telecom

## SETTING UP SECURITY SETTINGS

The following security parameters are available in eXpress:

- a limit of the number of requests for a specific IP address;
- filter by User-Agent;
- filter by DEF code;
- filter by phone number;
- a limit on the number of requests to a specific phone number.

### To set up security settings:

1. Go to the "Security" subsection.
2. Enter the values in the appropriate fields and click "Save" (see [Figure 24.](#)).

Request Rate Limiter by IP Adress

Maximum attempts

...per seconds

Save

Filter by User-Agent

User-Agent regex mask

example: ^Mozilla/5.?\$

Save

Filter by DEF-code

Int. code

example: +7

DEF-code list

example: 923,913

Save

Filter by phone

Phone regex mask

example: ^7923????\$

Save

Max. requests limit per phone number

Max. requests

Save

Unblock user

Phone or IP

example: 79090909090 || 127.0.0.1

Unblock

Figure 24.

## SETTING UP ADMINISTRATOR AUTHENTICATION

### To set up loading of administrator accounts from AD:

1. Go to the "Administrator Authentication" section.

A window will open (see [Figure 25](#)):

Figure 25

2. Configure the settings as shown in [Table 49](#).

The parameter values are provided by the Active Directory administrator.

Table 49

Parameter	Description
Address	Active Directory address
Port	AD connection port
Base DN	Directory object from which the search is performed
Search filter	<p>Filter for LDAP search.</p> <p>It shall ensure filtering of active users who are allowed to connect to this server.</p> <p>Recommended query construct:</p> <pre>"(&amp;(objectClass=person)(objectClass=user)(memberOf:1.2.840.113556.1.4.1941:=cn= express,ou=Groups,dc=firma,dc=local))"</pre> <p>where "cn= express,ou=Groups,dc=firma,dc=local" is the DN of the group, whose members will be eXpress users.</p> <p>When using cross-domain structures, specify the domain <b>DC=ru</b> in the connection parameters.</p> <p>An example of setting up synchronization of administrative users with a filter:</p> <pre>(!(memberOf=adm,OU=Groups,DC=example,DC=local)(memberOf=CN=adm_bot,OU=Groups,DC=example,DC=local)(memberOf=adm_ib,OU=Groups,DC=example,DC=local))</pre>
Administrator login	Login of the user who has read access to the list of users at the specified DN
Administrator password	Password of the user who has read access to the list of users at the specified DN
Password confirmation	Confirmation of the password of the user who has read access to the list of users at the specified DN

**To enable/disable authentication** of Active Directory administrators, check/uncheck "Enabled".

**To test the connection to Active Directory**, click "Test Connection".

After clicking on the "Show administrators" button, a list of Active Directory administrators is displayed.

## SETTING UP CONNECTIONS FOR CORPORATE SERVERS

The "Servers" section provides information about the RTS server to which the ETS server is connected (see [Figure 26](#)), and the CTS server connected to this ETS server (see [Figure 27](#)).

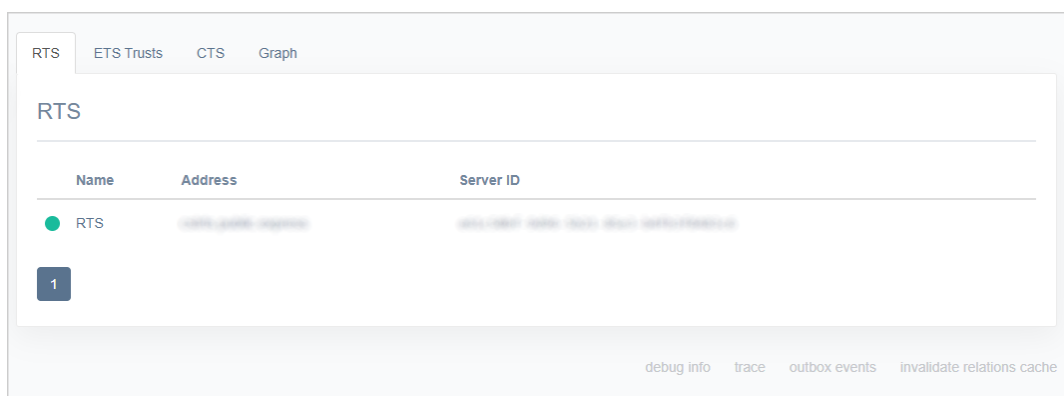


Figure 26

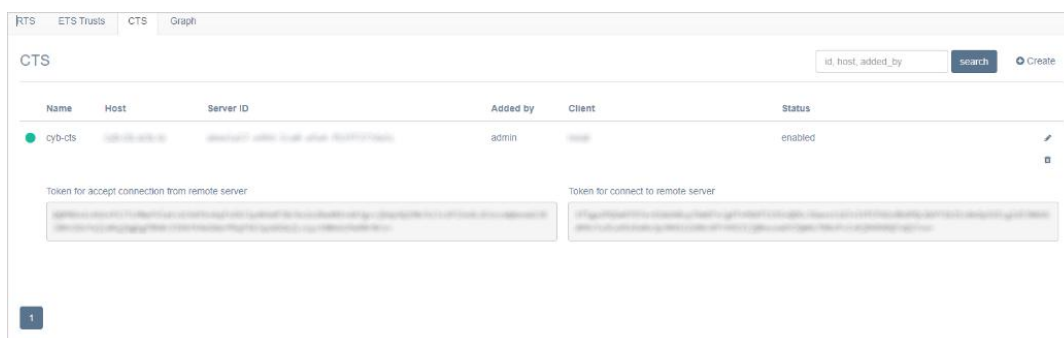


Figure 27

The connection status of the RTS and CTS servers is indicated by color markers next to the server names.

- green – the server is enabled and connection is available;
- purple – the server is locked out;
- red – the server is enabled and connection is not available;
- empty space – the server is connected to another RTS server.

The "Servers" section offers the following functionality:

- [viewing information about the graphical connection routing diagram](#);
- [viewing connection information for a single server on the graphical connection routing diagram](#).

**To view the graphical connection routing diagram**, open the "Graph" tab (see [Figure 28](#)).

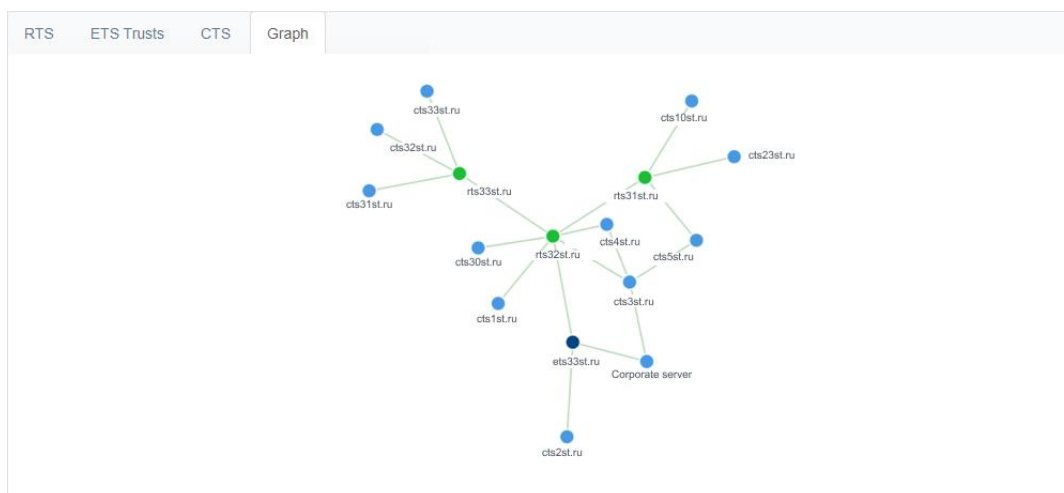


Figure 28

Servers are indicated in the diagram with colored circles, depending on the type:

- RTS – green;
- ETS – purple;
- CTS – blue.

For ease of viewing, diagram elements can be dragged with the mouse.

**To view information about connection to the server in the diagram:**

1. On the "Graph" tab, click the circle that represents this server.

The address of the selected server and the number of chats created on it will be displayed in the upper right corner of the screen (see Figure 29).

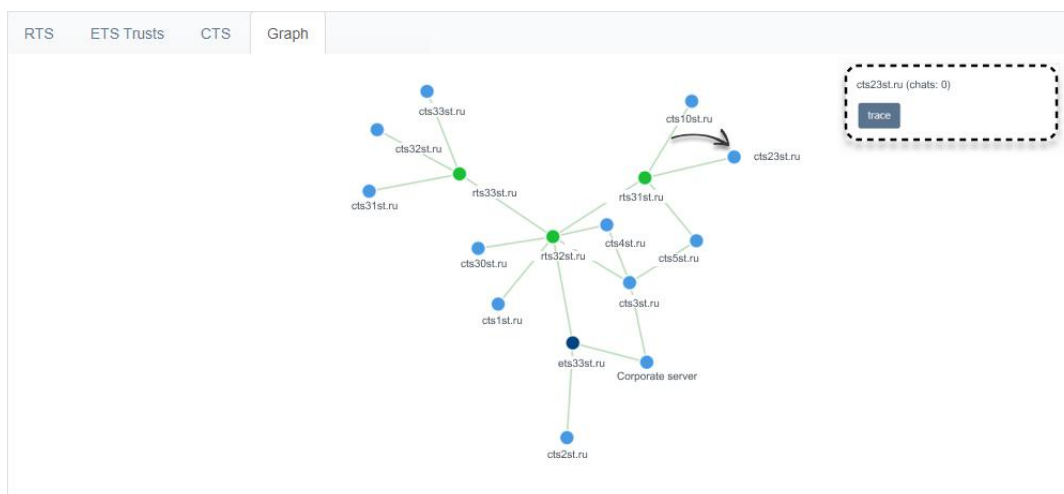


Figure 29

2. Click on the server name in the upper right corner of the screen.

A window will open with information about the RTS/ETS/TTS through which data is exchanged with the current server (see Figure 30).

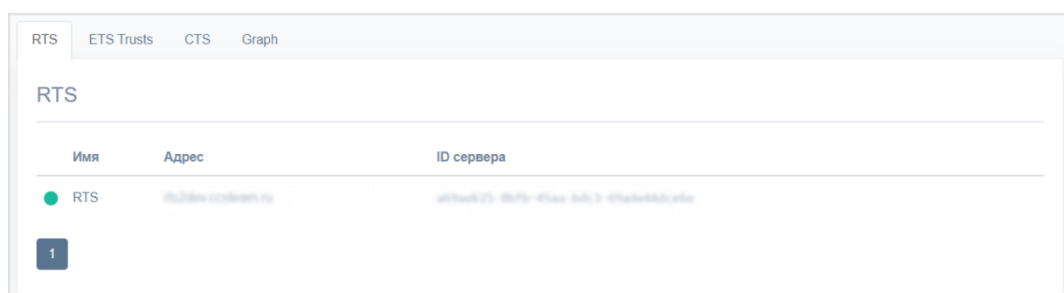


Figure 30

## SETTING UP CTS

The procedure for setting up the CTS server includes the following:

- [Connecting the TLS Certificate](#) (if this has not been done during the ETS installation process);
- [Connecting the Botx SSL certificate](#);
- [setting up video and voice communication](#);
- [Connecting the SMTP Server](#);
- [Connecting CTS Administrators from AD](#);
- [setting up integration with Active Directory](#);
- [setting up trusted connections](#).

## CONNECTING THE TLS CERTIFICATE AND THE BOTX SSL CERTIFICATE

### To use the TLS protocol in trusted connections:

1. Select the "Server" menu item.  
A window will open with information about this CTS server (see [Figure 31](#)).

Server Settings

Avatar

Choose File

No file chosen

Clear

Mobile background

Choose File

No file chosen

Clear

Mobile dark background

Choose File

No file chosen

Clear

Web background

Choose File

No file chosen

Clear

Web dark background

Choose File

No file chosen

Clear

Web high resolution background

Choose File

No file chosen

Clear

Web dark high resolution background

Choose File

No file chosen

Clear

☒ Hide server name

Save

Server Features

☐ Corporate search
 ☐ Trust search
 ☐ Enable e2e encryption by default in group chats
 ☐ Enable e2e encryption by default in channels
 ☐ Disable corporate phonebook

☒ Allow the user to control their avatar
 ☒ Moderate user's requests for profile changes

Save

Notification during logging in

☒ Show the user when logging in  
Whether to show the document to the user when they first log in

Consent, ru

Choose File

No file chosen

View

Consent, en

Choose File

No file chosen

View

Save

Notification of technical works

☐ Enabled  
 Technical work is underway, there may be some problems with the app.

Under maintenance alert, ru

Ведутся технические работы, возможны перебои в

Under maintenance alert, en

Technical work is underway, there may be some probl

Save

Set default alert text.

Notification about update

The user will get a notification if an app update is available.
 ☒ Notify about an existing update
 ☒ Block application interface until update is started

Versions late

0

Enabled for: ☐ IOS ☐ Android ☒ Desktop

Save

RTS ID

production-3360-ru-ru-3360-3360-3360-3360

CTS ID

production-3360-ru-ru-3360-3360-3360-3360

Trusts TLS Certificate

Certificate

Certificate is not present

Choose File

N...en

Key

Choose File

N...en

Save

BotX SSL Certificate

Certificate

Certificate is not present

Choose File

N...en

Save

Admin Info

Full name

Александр Троянов

Phone

Address

Emails (comma separated)

Save

Service versions

ad\_integration 3.36.1

admin 3.36.0

botx 3.36.0

email\_notifications 3.36.0

kdc 3.36.0

messaging 3.36.1

phonebook 3.36.0

file\_service 3.36.3

routing\_schema 3.36.0

settings 3.36.0

trusts 3.36.1

voex 3.36.0

metrics\_service 3.36.0

corporate\_directory 3.36.0

Figure 31

2. Enter the information about the certificate and the key in the appropriate fields in the Trust TLS Certificate area (see [Figure 32](#)).

Figure 32

3. Click "Save".

**Note.** It is allowed to use the TLS certificate used during the CTS server installation stage.

**To connect the chatbot certificate,** in the "BotX SSL certificate" area, enter the certificate information and click "Save" (see [Figure 33](#)).

Figure 33.

## SETTING UP VIDEO AND VOICE COMMUNICATION

Setting up video and voice communication is performed after installing the Media server and is described on page [59](#).

## CONNECTING THE SMTP SERVER

### To connect the SMTP Server:

1. Select "E-mail" from the menu.  
The "E-mail Settings" window will open for entering parameters (see [Figure 34](#)):



Figure 34

2. In the "E-mail Settings" window, fill in the fields as follows (see [Table 50](#)):

Table 50

Field	Description
App name	The name of the app from which e-mails will be sent
From	Return address
Server	FQDN or IP address of the mail server
Port	Port number for retransmission of outgoing mail: 25, 587 or 465. The port number depends on the type of connection
User name	E-mail address
Password	Data for authorization on the SMTP server. If authentication on the mail server is not used, then leave these fields blank
Password confirmation	Data for authorization on the SMTP server. If authentication on the mail server is not used, then leave these fields blank
Connection protection	Type of secure connection (drop-down list: SSL, Start/TLS or empty value)
Send e-mail via	Drop-down list for selecting a server from which e-mails will be sent (if you select "Local settings" in the drop-down list, e-mails will be sent via the server configured in this window; if you select "RTS", e-mails will be sent via RTS).

3. Click "Save".

**To check connection settings**, use the "Test E-mail Sending" area. Enter the recipient's address in the empty field and click "Send".

## SETTING UP ADMINISTRATOR AUTHENTICATION

This section is intended for connecting administrators using AD.

### To set up loading of administrator accounts from AD:

1. Go to the "Administrator Authentication" section (see [Figure 35](#)).

Administrators authentication

Address

Port

Base DN

Search filter

Administrator login

Administrator password

Password confirmation

☒ Enabled

Save

Check connection

Show administrators

Figure 35

- Configure the settings as shown in [Table 51](#).

The parameter values are provided by the Active Directory administrator.

Table 51

Parameter	Description
Address	Active Directory address
Port	AD connection port
Base DN	Directory object from which the search is performed
Search filter	<p>Filter for LDAP search.</p> <p>It shall ensure filtering of active users who are allowed to connect to this server.</p> <p>Recommended query construct:</p> <p>"(&amp;(objectClass=person)(objectClass=user)(memberOf:1.2.840.113556.1.4.1941:=cn= express,ou=Groups,dc=firma,dc=local))" where "cn= express,ou=Groups,dc=firma,dc=local" is the DN of the group, whose members will be eXpress users.</p> <p>When using cross-domain structures, specify the domain <code>DC=ru</code> in the connection parameters.</p> <p>An example of setting up synchronization of administrative users with a filter:</p> <p>(!(memberOf=adm,OU=Groups,DC=example,DC=local)(memberOf=CN=adm_bot,OU=Groups,DC=example,DC=local)(memberOf=adm_ib,OU=Groups,DC=example,DC=local))</p>
Administrator login	Login of the user who has read access to the list of users at the specified DN
Administrator password	Password of the user who has read access to the list of users at the specified DN
Password confirmation	Confirmation of the password of the user who has read access to the list of users at the specified DN

**To enable/disable authentication** of Active Directory administrators, check/uncheck "Enabled".

**To test the connection to Active Directory**, click "Test Connection".

After clicking on the "Show administrators" button, a list of Active Directory administrators is displayed.

## SETTING UP REGISTRATION

**Important!** An unsuccessful combination of customization of the first login screen of the app, enabling and disabling registration without a number and the ability to set the user's rights to perform a number of operations with their phone number can lead to an unsuccessful combination that will cause loss of access to the app! A number of common errors caused by incorrect server configuration are described in the section [“Troubleshooting Typical Errors”](#).

The following methods are available to the administrator for setting up user registration/authorization in the system:

- [Active Directory \(NTLM\)](#);
- [E-mail](#);
- [OpenID](#);
- [Registration Without Telephone Number](#).

### To select a registration method:

1. Go to the “Registration Settings” section and select “OpenID” (see [Figure 36](#)).
2. Select the registration method.
3. Select a synchronization source.
4. Set the synchronization schedule (in the cron format).
5. Click “Save”.

Registration Settings

CTS-only registration

☒ Allow registration without phone number  
If enabled user can log in without confirmation by SMS-code

☐ Display a warning that registration will be forbidden soon

☐ Allow skipping two-factor authentication within a contour  
If enabled user can log in account with assigned phone within a contour without confirmation by SMS-code

User's contact management

☒ Allow to add phone number  
☒ Allow to edit phone number  
☒ Allow to delete phone number

Registration methods

E-mail NTLM **OpenID**

Synchronization sources

NTLM OpenID **None**

Full Synchronization schedule (cron format)

0 23 \* \* \*

example: 0 23 \* \* \*

Sync synced users: 44 / with e-mail: 26

Save

Figure 36

The selected registration method will be saved. A corresponding system message will be displayed at the top of the screen.

**To complete the configuration**, set the parameters for the specified method in the corresponding tab: E-mail, NTML or OpenID.

## SETTING UP INTEGRATION WITH ACTIVE DIRECTORY

**To integrate with AD**, connect to AD and upload contacts to the server.

When integrating eXpress with a Microsoft Active Directory-based corporate directory, it is necessary to create an account with "Domain Users" rights and a "Deleted Objects" container (<https://support.microsoft.com/en-us/help/892806/how-to-let-non-administrators-view-the-active-directory-deleted-object>).

### To connect to Active Directory:

**Note.** To correctly configure the system for the customer's domain, it is recommended to involve an Active Directory administrator.

1. Go to the "Active Directory" section.

A window for setting up registration parameters via Active Directory will open (see [Figure 37](#)).

2. In the left column, in the text fields, specify the parameter values for synchronizing LDAP users (see [Figure 37](#), [Table 52](#)).

Figure 37

Table 52

Parameter	Purpose
IP address	VDAP IP address; If LDAPS connection is required, enter "ldaps://" before the domain name or IP address, e.g. "ldaps://firma.local"
Port	AD connection port. For the LDAP protocol, enter the value "389", for the LDAPS protocol, enter the value "636"
Domain	The domain of the server to which the accounts are uploaded
Base DN	Directory object from which the search is performed
Search filter	Filter for searching in Active Directory
Search attributes for exporting	Exported account attributes
User login for synchronizing LDAP users	Login for connecting to AD for synchronization
User password for synchronizing LDAP users	Password for connecting to AD for synchronization
Password confirmation	Confirmation of password for connecting to AD for synchronization
Number of incorrect input attempts before lockout	The maximum number of password attempts after which the account is locked out
Lockout timeout when entering incorrect password (in seconds)	The time interval, in seconds, during which the application is locked out when an incorrect password is entered

- Specify events in Active Directory that will cause the eXpress user to be re-prompted to authenticate to the corporate eXpress server (see Table 53):

Table 53

Parameter	Purpose
Logoff by disabled	After disabling a user account, creates a request to disconnect the user from the CTS server. This request requires confirmation in the "Logout Requests" section, after confirmation the user will be automatically disconnected from the CTS server.
Logoff by lockout	Once the user account is temporarily locked out in AD, a request is generated to disconnect the user from the CTS server. All active user sessions will be closed. This request requires confirmation in the "Logout Requests" section, after confirmation the user will be automatically disconnected from the CTS server.
Logoff by password expired	If the user's AD password has expired, a request is created to log the user off from the CTS. All active user sessions will be closed. This request requires confirmation in the "Logout Requests" section, after confirmation the user will be automatically disconnected from the CTS server.
Logoff by account expired	If the user's AD account has expired, a request is created to disconnect the user from the CTS server. All active user sessions will be closed. This request requires confirmation in the "Logout Requests" section; after confirmation the user will be automatically disconnected from the CTS server
Automatic logout when excluded from AD synchronization selection	If the user account is removed from a group, a request is created to disconnect the user from the CTS server. All active user sessions will be closed. This request requires confirmation in the "Logout Requests" section; after confirmation the user will be automatically disconnected from the CTS server
Logoff by password change	If the user's AD account password has been changed, a request is created to disconnect the user from the CTS server. All active user sessions will be closed. This request requires confirmation in the "Logout Requests" section; after confirmation the user will be automatically disconnected from the CTS server
User Account Disabled (AD LDS)	If the user account is locked, a request is created to disconnect the user from the CTS server. All active user sessions will be closed. This request requires confirmation in the "Logout Requests" section; after confirmation the user will be automatically disconnected from the CTS server

- In the right column, specify the attributes that will be displayed in the user card. This setting is described in more detail below.

5. Select the authorization method: simplified or via NTLM, by clicking on the "Simple" or the "NTLM" button.
6. Click "Save" to save your changes.  
If all settings are specified correctly, the list of users will be displayed in the "Users" section within three hours.

**To perform synchronization with LDAP**, click "Synchronize".

**To delete changes**, click "Delete".

In the event that any problems occur during synchronization, check the correctness of the data received from AD using the `ldapsearch` command (parameters that need to be replaced in accordance with the AD connection settings are highlighted in red):

```
$ ldapsearch -v -h myhost.mydomain.mytld -p 389 -D 'mydomain\myuser'
-W -b "cn=Users,dc=mydomain,dc=mytld" -s sub
"(&(objectCategory=person)(objectClass=user)(memberOf:1.2.840.113556.1.4.1941:=CN=ExpressUsers,CN=Users,DC=mydomain,DC=mytld))" -x
```

**Note.** For Ubuntu version 19 or higher, and if the error occurs in other operating systems, run the following command:

```
$ ldapsearch -v -H myhost.mydomain.mytld -p 389 -D 'mydomain\myuser' -W -b
"cn=Users,dc=mydomain,dc=mytld" -s sub
"(&(objectCategory=person)(objectClass=user)(memberOf:1.2.840.113556.1.4.1941:=CN=ExpressUsers,CN=Users,DC=mydomain,DC=mytld))" -x
```

**To provide user access to eXpress**, create an eXpress user group in Active Directory. The group type is – "Security" and the group visibility is "Universal".

When integrating eXpress with a corporate directory based on an LDAP-compatible server, create an account with directory read access rights.

**To set up the visibility of profile fields:**

1. Go to the "Field Visibility Settings" section.  
The "Profile Fields Visibility" window will open (see [Figure 38](#)):

Profile fields visibility

Public name  
share to all users

Full name  
share to all users

Company  
share to all users

Position  
share to corporate users only

Department  
share to corporate users only

Avatar  
share to corporate users only

Phone  
share to corporate users only

Telephone Number (Other)  
share to corporate users only

IP Phone  
share to this CTS only

IP Phone (Other)  
share to this CTS only

E-mail  
share to all users

Description  
share to corporate users only

Office  
share to trusted CTS only

Manager  
share to trusted CTS only

Other ID  
share to this CTS only

Save

Figure 38

- Set the values of corporate profile variables in the access fields.

Corporate profile variables are automatically populated with values from the AD database and are available for viewing in the application in the chat card. A description of the data access level is provided in [Table 54](#):

Table 54

Field name	Comment
No one	The value of this field is not available for viewing in the application
Only for users from the same CTS server	The value of this field is available for viewing in the application only to users registered on this corporate server
Only for users from trust CTS	The value of this field is available for viewing in the application only to users who are registered: <ul style="list-style-type: none"> <li>on this corporate server;</li> <li>servers with which a trust connection is established</li> </ul>
For corporate users only	The value of this field is available for viewing in the application to all users registered in the corporate contour
For all	The value of this field is available for viewing in the application to all users

- Click "Save".

The configured fields will become available to the specified users. The system message "Profile fields visibility settings saved" will be displayed at the top of the screen.

## SETTING UP E-MAIL

### To set up registration using an e-mail mask:

1. Go to the "Registration Settings" tab and select "E-mail".  
The "E-mail Settings" window will open (see [Figure 39](#)).
2. Enter the E-mail mask in the field using a regular expression (for example: `^.*?@corporate.local`).
3. Click "Save".

Figure 39

After successfully saving the changes, the system message "Registration by e-mail mask settings saved" will be displayed at the top of the screen.

## SETTING UP OPENID

**Note.** Before setting up OpenID, it is necessary to set up integration between the CTS server and Keycloak. See ["CTS and Keycloak Integration"](#).

### To set up OpenID:

1. Go to the "Registration Settings" section and select "OpenID".  
A window for setting up registration parameters via OpenID will open (see [Figure 40.](#)).
2. Fill in the fields as follows (see [Table 55](#)):

Table 55

Field	Description
OpenID provider	Switch to auto-substitute the /auth prefix in keycloak requests. For versions below 17, a prefix is added
OpenID host	URL at which Keycloak is available (includes mandatory protocol), for example — <a href="https://openid.provider.com">https://openid.provider.com</a>
OpenID port	The port on which Keycloak accepts requests, for example 8443
OpenID Realm ID	The name of the realm to which CTS will be connected, for example — Express
OpenID client ID	The name of the OpenID Connect client that CTS will access, for example — cts-adintegration
OpenID client Secret	The client secret key specified in the Keycloak credentials menu, for example — aNicQoU5k8UK7BZsUJYaegT493e8pYaX
OpenID redirect URI	The CTS URI that the browser will redirect users to after a successful login, for example —



Field	Description
	<a href="https://cts.express/api/v1/ad_integration/openid/success">https://cts.express/api/v1/ad_integration/openid/success</a>
Possible OpenID Redirect URIs	The list of URL addresses of WEB clients from which redirection is allowed (separated by commas). Only used when disabling iframe windows in Web/Desktop client apps. For example — <a href="https://web-beta.express">https://web-beta.express</a> , <a href="https://web.express">https://web.express</a>
OpenID response type	The value of the OpenID Connect response_type parameter. The value is always specified as "code"
OpenID scope	A space-separated list of scopes that are requested using the scope parameter, for example — openid express-scopes email offline_access roles
OpenID role required	Specifies the name of the user role allowed to log into CTS, for example — user_cts01
Path to the list of roles. Use dot for nested paths, e.g. "path.role"	The path to the list of roles, for example — realm_access.roles
Asynchronous update response timeout (in milliseconds)	Waiting time for response from Keycloak, for example — 5,000
OpenID login prefill	Enables the selection of the login field filling mode in the Keycloak login form
Device authorization method	Type of authorization of client applications by QR code, for example — CIBA

OpenID Registration Settings

OpenID provider

KeyCloak < 17

KeyCloak ≥ 17

Blitz

OpenID host

example: https://openid.provider.com

OpenID port

OpenID realm ID

OpenID client ID

OpenID client secret

OpenID redirect URI

OpenID valid redirect URIs

OpenID response type

OpenID scope

OpenID required role

OpenID role path. Use dot notation for nested path "path.role"

realm\_access.roles

OpenID async timeout(in ms)

5000

OpenID login prefill

Don't prefill

▼

☒ Logout by disabled  
☐ Logout by missing user role

Device authorization method

CIBA

Device Auth Flow

UserID

user\_id

Public name

Full name

Username

Domain

Company

Position

Department

Avatar

Phone

Telephone Number (Other)

IP Phone

IP Phone (Other)

E-mail

Description

Office

Manager

Other ID

Personnel Number

Business Unit

Personnel Category

Gender

Birthday

Save

Figure 40.

**Note.** In the “OpenID scope” field, it is recommended to specify the value from the “scope” line of the Keycloak administrator console. This is necessary to obtain the “scope” list being transmitted.

To do this, open the Keycloak administrator console, go to the “Clients” section → Client scopes → Client ID → Evaluate → Generated access token → “scope” line and copy the value (see [Figure 41](#)).

3. In the fields of the right column, specify the attributes that will be displayed in the user card.
4. Click “Save”.

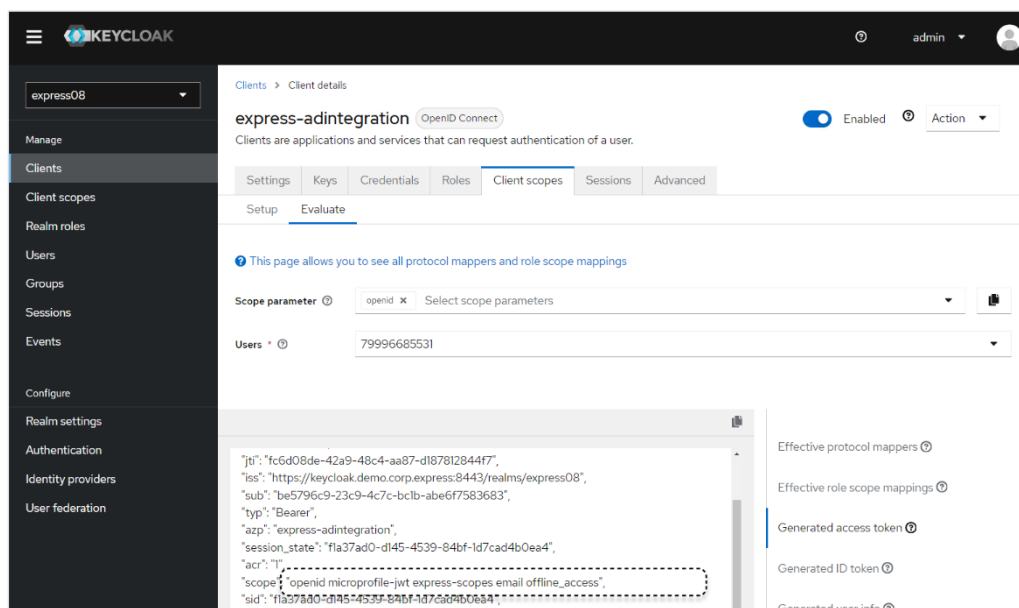


Figure 41

## REGISTRATION WITHOUT TELEPHONE NUMBER

### To set up registration without telephone number:

1. Go to the “Registration Settings” section.
2. Check/uncheck the box “Registration without telephone number is allowed” (the setting is enabled by default).
3. Check/uncheck the box “Display a warning about imminent disconnection” (can be changed only if registration without telephone number is allowed).
4. Check/uncheck the box “Allow login without SMS code confirmation within the contour” (the setting is disabled by default).
5. Set permissions for the user to operate with the phone number (all permissions are granted by default).
6. Click “Save”.

## SETTING UP TRUSTED CONNECTIONS

### To create a trust connection:

1. Select the “Server” menu item.
2. Go to the “Trusts” tab (see [Figure 42](#)).

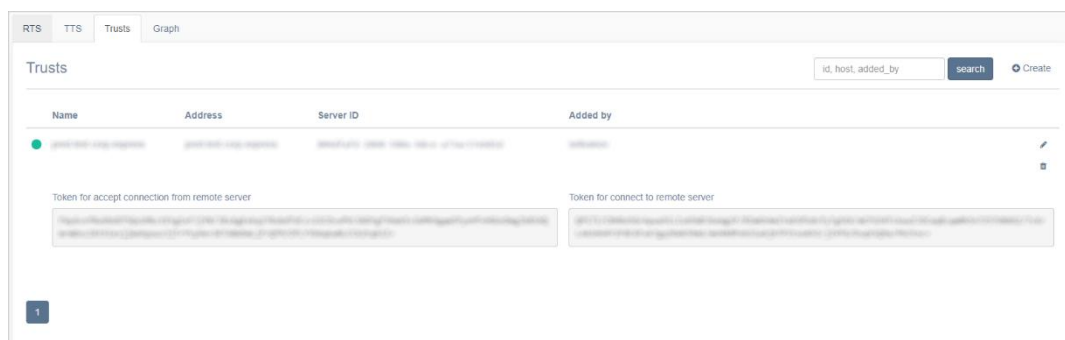


Figure 42

- Click "Create" in the upper right corner.  
A window will open (see Figure 43):

Create trust

Back to list

Remote CTS ID

Name

Token for accept connection from remote server

Token for connect to remote server

Endpoint

Connection config

App gateway url

Transport encryption (choose from: tls, tls\_probe, libsodium or leave blank)

☐ App gateway enabled

☐ Allow trust search

Save

Figure 43

- Fill in the fields as follows (see Table 56):

Table 56

Field	Description
CTS ID	ID of the CTS server with which the connection will be established. The CTS server ID is stored in the "Server" menu item in the administrator web interface of this server
Name	Short designation for the trust to be created
Token for accepting connection from a remote server	Token name
Token for connection to a remote server	Token name
Endpoint	Server connection address. In the table with the list of tokens, the data from this field is displayed in the "Address" column

Field	Description
Allow trust search	Allows another server to access the corporate contact book of the server on which the trust is created. Trust search is available if Corporate Search is enabled in the server settings

---

**Example:** You need to create a trust between two servers: CTS1 and CTS2. To solve this problem, the administrator creates a trust on each server, specifying tokens in the settings so that the token for connecting on the CTS1 server matches the token for receiving a connection on the CTS2 server, and vice versa.

---

5. Click "Save".

Next, go to the administrator web interface of the corporate server (in the example given in step 2, CTS2) with which the connection is being established, and create a trust with the current server (CTS1).

## Chapter 4

### UPDATE PROCEDURE

The complete procedure for updating the system, its components and additional software is described in the document "Administrator's Guide. Updating".

The system update procedure includes the following:

- updating the operation system;
- updating the servers manually;
- updating the servers with the use of Ansible scripts;
- updating a fault-tolerant configuration;
- updating the Media server.

The procedure for updating additional system components and integration software includes the following:

- updating Desktop version;
- updating the certificate;
- updating PostgreSQL.

The document "Administrator's Guide. Updating" contains a description of the procedure for updating eXpress CS to version 3.27 with a change in application architecture, and the process of migrating large databases.

The document also provides a description of possible emergency situations when updating from the local Registry repository.

## Chapter 5

### TROUBLESHOOTING TYPICAL ERRORS

**Note.** All operations on the servers shall be carried out on behalf of the superuser.

eXpress CS supports the following message types (see [Table 57](#)):

*Table 57*

Error message	Value
403 Forbidden — You don't have access to view this page	The administrator does not have access rights
404 Page Not Found	The page is missing
413 Request is too large	Occurs if the administrator attempts to upload a file that is too large, for example, an avatar
500 Internal Server Error	Exceptional error

**To obtain superuser rights**, run the following commands:

```
sudo -s
```

eXpress CS is built on a microserver architecture using containerization based on Docker software. In eXpress CS, all maintenance operations and troubleshooting operations are performed with Docker containers.

In case of problems in the operation of eXpress CS, first of all it is necessary to check the operation status of the containers.

**To check the status of containers ("Up" or "Exited")**, run the following command:

```
docker ps -a --format "{{.Names}}: {{.Status}}"
```

The normal status of containers is "UP".

If a container has the "Exited" status, start it with the following command:

```
docker start <container name in the cts-containername_1 format>
```

If the problem has not been resolved, collect system logs.

**To collect logs, run the following command:**

```
cd /opt/express
dpl --dc logs --tail=1000 > logs.txt
```

Send the collected logs to the administrators responsible for eXpress CS.

If the user cannot log into the server, collect logs with the following command:

```
cd /opt/express
dpl --dc logs --tail=1000 ad_integration > logs.txt
```

**To restart all containers**, run the following command:

```
cd /opt/express
dpl --dc restart
```

If users have a problem with the order in which messages are displayed in conversations, check the time on the server with the following command:

```
date
```

If the time is incorrect, check the status of the chronyd time service.

**To check the status of the time service**, run the following command:

```
systemctl status chronyd
```

If the status "active" has the value of "inactive", start the service with the following command:

```
systemctl start chronyd
```

### Authorization Error

This error may be displayed if a user account is created in the system after synchronization with AD (see [Figure 44](#)):

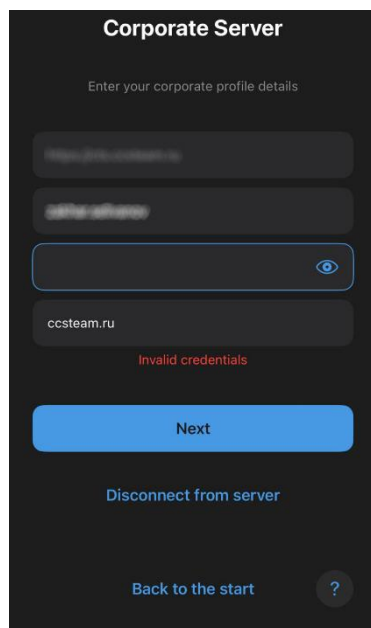


Figure 44

This error occurs if users with different User logon names are registered in AD and a different domain is specified in the AD settings.

To solve the authorization problem, the user must supplement his login with a domain with the use of the @ symbol, for example [user9@it-company.local](#).

[Table 58](#) shows possible errors that may occur in the event of an incorrect combination of user registration settings:

Table 58

No.	Description of the combination of settings	Description of consequences
1.	The "phone and credentials" button is disabled in the build, the "registration without number is allowed" option is disabled in the administrator web-interface	Thus, both registration methods are prohibited. Only those users who have previously added their telephone number to their profile via a redirect to SMS input will be able to access the app
2.	All buttons except for "Phone and Credentials" button are disabled in the build, and adding the telephone number is disabled in the administrator web interface	Users who have not added a number to their profile will not be able to log into the app
3.	In the administrator web interface, registration without the telephone number has been disabled without prior notification to users about the need to add their telephone number	The users who have not added their telephone number to their profile will not be able to log into the app



No.	Description of the combination of settings	Description of consequences
4.	Users were initially registered on a public server, after which they were asked to log in using their credentials via "Corporate E-mail" or "Corporate Server Address"	In this case, the users have two accounts: a public one with the telephone number and a corporate one without the telephone number. It is technically impossible to combine the two into one. The only way out of this situation is to add the same telephone number to the corporate account and delete the public account, losing all correspondence.
5.	In the branded build, all buttons are disabled except for "Phone and Credentials", and in the administrator web interface, the users are allowed to delete their telephone number	A user who has deleted their telephone number will not be able to log into the app
6.	In the administrator web interface, registration without the telephone number is prohibited, but the users are allowed to delete their telephone number	A user who has deleted their telephone number will not be able to log into the app

## Chapter 6

### ELIMINATING VULNERABILITIES

#### To eliminate the log4j (CVE-2021-44228), CVE-2021-45046 vulnerability:

**Note.** If the version is less than 2.16, but not 1.x, then an update to the latest version is required. In versions 1.x, this vulnerability is not present.

1. Check the Log4j version using the following command:

```
find / -name 'log4j*.jar'
```

or find it via the CLASSPATH output of your java installation:

```
echo $CLASSPATH
```

2. In the Java Virtual Machine (JVM) settings for Log4j packages version 2.0 to 2.15, add the following flag for the application:

- Dlog4j2.formatMsgNoLookups=true;

**Important!** Install the latest Log4j 2.16.0 service pack, which fixes the service pack that was made for CVE-2021-45046 Log4j 2.15.0. Installing the latest package is no different from the previous installation and is not required unless you are using the additional APM program with logging set to “tracing” mode.

Otherwise, it is highly recommended to upgrade your current Elasticsearch version to 7.16.1 (or 6.8.21) and perform a sequential restart of your nodes.

- for elasticsearch — /etc/elasticsearch/jvm.options;
- for logstash — /etc/logstash/jvm.options.

**Note.** The path may differ and depends on the installation method.

3. Restart the app using the following command:

```
systemctl restart elasticsearch
```

4. Check that the jvm setting is active:

```
ps axw | grep formatMsgNoLookups
```

The flag should be visible in the app launcher.

Example of Log4j update for Elasticsearch:

```
wget https://d1cdn.apache.org/logging/log4j/2.16.0/apache-
log4j2.16.0-bin.tar.gz
tar zxvf apache-log4j-2.16.0-bin.tar.gz
cd apache-log4j-2.16.0-bin/
ls /usr/share/elasticsearch/lib/log4*
cp log4j-api-2.16.0.jar /usr/share/elasticsearch/lib/
cp log4j-core-2.16.0.jar /usr/share/elasticsearch/lib/
rm -f /usr/share/elasticsearch/lib/log4j-api-2.11.1.jar
rm -f /usr/share/elasticsearch/lib/log4j-core-2.11.1.jar
```

For Logstash versions earlier than 6.8.21 and 7.16.0 run the following commands:

```
zip -q -d <LOGSTASH_HOME>/logstash-core/***/log4j-core-2.*  
org/apache/logging/log4j/core/lookup/JndiLookup.class
```

or

```
wget https://dlcdn.apache.org/logging/log4j/2.16.0/apache-log4j-  
2.16.0-bin.tar.gz  
tar zxvf apache-log4j-2.16.0-bin.tar.gz  
cd apache-log4j-2.16.0-bin/  
ls /usr/share/logstash/lib/log4j*  
cp log4j-api-2.16.0.jar /usr/share/logstash/lib/  
cp log4j-core-2.16.0.jar /usr/share/logstash/lib/  
rm -f /usr/share/logstash/lib/log4j-api-2.11.1.jar  
rm -f /usr/share/logstash/lib/log4j-core-2.11.1.jar
```

# Appendix 1

## SINGLE CTS NETWORK INTERACTIONS

No.	Source	Destination	Port and protocol	Description
1	Single CTS server	Bot server	TCP/8000-8100 TCP/443	Interaction of Single CTS with Bot server, interaction of Bot server with Single CTS via the HTTP/HTTPS protocol
	Bot server	Single CTS server		
2	Internal IS	Bot server	TCP/80 TCP/443 TCP/8000-8100	Interaction of internal information systems with the Bot server, interaction of the Bot server with internal information systems via the HTTP/HTTPS protocol
	Bot server	Internal IS		
3	Single CTS server	LDAP server	TCP/389, 636	Ensuring the operation of LDAP/LDAPS
4	Administrator	Single CTS and Media server	TCP/22	Server administration via the SSH protocol
			TCP/443	eXpress administration via web interface and the HTTPS protocol
5	Single CTS server	SMTP server	TCP/25 TCP/587 TCP/465	Ensuring the sending of e-mails with authentication PIN-code via the SMTP protocol
6	Single CTS server	DNS and NTP server	TCP/53 UDP/53	Ensuring DNS name resolution
			UDP/123	Ensuring the operation of the NTP service
7	Single CTS server	Media server	TCP/8188	Providing authentication and encryption for voice calls
8	Transcoding	Single CTS server	TCP/443	Transferring transcoded recordings to file storage
9	Transcoding	Media server	TCP/443	Ensuring the transfer of call recording files via HTTPS protocol for subsequent processing by the Transcoding server
10	Internal user	Single CTS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
11	Internal user	Media server	UDP/20000-40000	Ensuring media data transmission via the SRTP conferencing protocol
12	External user	Media server (External IP NAT)	TCP/3478 UDP/3478	Ensuring the operation of the STUN/TURN protocols
			UDP/20000-40000	Ensuring media data transmission via the SRTP conferencing protocol
13	External user	Single CTS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
14	Single CTS server	Let's Encrypt servers (ANY)	TCP/80 TCP/443	When using a free certificate from Let's Encrypt
	Let's Encrypt servers (ANY)	Single CTS server		
15	Single CTS server	Installation and update server registry.public.ex press	TCP/443	Access to the Docker image repository for installing and updating the eXpress software

No.	Source	Destination	Port and protocol	Description
16	Single CTS server	eXpress CTS partner server	TCP/5001	Ensuring direct message communication between corporate servers bypassing the public contour.
	eXpress CTS partner server	Single CTS server		
17	Media server	eXpress CTS partner server	UDP/20000-40000	Ensuring media data transmission via the SRTP protocol
	eXpress CTS partner server	Media server		
18	External user	RTS ru.public.express	TCP/443	Ensuring the interaction of an external user with the RTS server
19	Single CTS server	RTS ru.public.express	TCP/5001	Ensuring the interaction of the corporate eXpress server and the RTS server
20	Internal user	RTS ru.public.express	TCP/443	Client access to the eXpress public contour via the HTTPS protocol
21	External user	corp.express Web Client server	TCP/443	Client access to the web client server in the public contour
22	Internal user	corp.express Web Client server	TCP/443	Client access to the web client server in the public contour
23	RTS ru.public.express	SMS Operator	TCP/443	Sending text messages (SMS) to users
24	RTS ru.public.express	Huawei push notification service	TCP/443	Sending push notifications to Huawei users
25	RTS ru.public.express	Apple push notification service	TCP/443	Sending push notifications to iOS users
26	RTS ru.public.express	Google push notification service	TCP/80	Android push notifications to Android users

The following ports and protocols must be NAT IP-to-IP configured and translated for the Single CTS:

- TCP/443 (including for the Media server);
- TCP/5001;
- TCP/3478 (only for the Media server);
- UDP/3478 (only for the Media server);
- UDP/20000-40000 (only for the Media server).

TCP/80 port is added when using Let's Encrypt.

## Appendix 2

### FRONT CTS, MEDIA AND BACK CTS NETWORK INTERACTIONS

No.	Source	Destination	Port and protocol	Description
<b>Main Network Interactions</b>				
1	Back CTS server	Bot server	TCP/8000-8100 TCP/443	Interaction of Back CTS with Bot server, interaction of Bot server with Back CTS via the HTTP/HTTPS protocol
	Bot server	Back CTS server		
2	Internal IS	Bot server	TCP/8000-8100 TCP/80 TCP/443	Interaction of internal information systems with the Bot server, interaction of the Bot server with internal information systems via the HTTP/HTTPS protocol
	Bot server	Internal IS		
3	RTS ru.public.express	Google push notification service	TCP/443	Android push notifications to Android users
4	Back CTS server	Front CTS server	TCP/8888	Tinyproxy local proxy server for connection of Back CTS to the repository of Docker images used to install and update the product
			TCP/443	Monitoring of trusts container operation
5	Front CTS server	Back CTS server	TCP/443	Transmission of encrypted user data with TLS transport wrapper
			TCP/2379	Connection to the configuration storage to retrieve various service settings
			TCP/5432	Connection of the trusts container to the database for storing information necessary for operation
			TCP/9092	Connection to the Kafka software message broker to exchange events between services
			TCP/6379	Connection to Redis for the caching function
6	Administrator	Front CTS, Back CTS and Media servers	TCP/22	Server administration via the SSH protocol
			TCP/443	eXpress administration via web interface and the HTTPS protocol
7	Back CTS server	SMTP server	TCP/25 TCP/587 TCP/465	Ensuring the sending of e-mails with authentication PIN-code via the SMTP protocol
8	Back CTS server	Media server	TCP/8188	Providing authentication and encryption for voice calls
9	Transcoding	Media server	TCP/443	Ensuring the transfer of call recording files via HTTPS protocol for subsequent processing by the Transcoding server
10	Transcoding	Back CTS server	TCP/443	Transferring transcoded recordings to file storage
11	Internal user	Back CTS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
12	Front CTS server	DNS and NTP server	TCP/53 UDP/53	Ensuring DNS name resolution
			UDP/123	Ensuring the operation of the NTP service
13	Internal user	Media server	UDP/20000-40000	Ensuring media data transmission via the SRTP conferencing protocol
14	External user	Media server	TCP/3478	Ensuring the operation of the STUN/TURN protocols

No.	Source	Destination	Port and protocol	Description
		(External IP NAT)	UDP/3478	
			UDP/20000-40000	Ensuring media data transmission via the SRTP protocol
15	External user	Front CTS server (External IP NAT)	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
16	Front CTS server	Installation and update server registry.public.express	TCP/443	Access to the Docker image repository for installing and updating the eXpress software
17	External user	RTS ru.public.express	TCP/443	Ensuring the interaction of an external user with the RTS server
18	Front CTS server	RTS ru.public.express	TCP/5001	Ensuring the interaction of the corporate eXpress server with the RTS server
19	Internal user	RTS ru.public.express	TCP/443	Client access to the eXpress public contour via the HTTPS protocol
20	External user	corp.express web client server	TCP/443	Client access to the web client server in the public contour
21	Internal user	corp.express web client server	TCP/443	Client access to the voice communication server in the public contour
22	Front CTS server	Let's Encrypt servers (ANY)	TCP/443 TCP/80	When using a free certificate from Let's Encrypt
	Let's Encrypt servers (ANY)	Front CTS server		
23	Front CTS server	eXpress CTS partner server	TCP/5001	Ensuring direct message communication between corporate servers bypassing the public contour
	eXpress CTS partner server	Front CTS server		
24	Media server	eXpress CTS partner server	UDP/20000-40000	Ensuring media data transmission via the SRTP protocol
	eXpress CTS partner server	Media server		
25	RTS ru.public.express	SMS operator	TCP/443	Sending text messages (SMS) to users
26	RTS ru.public.express	Huawei push notification service	TCP/443	Sending push notifications to Huawei users
27	RTS ru.public.express	Apple push notification service	TCP/443	Sending push notifications to iOS users
<b>Authentication with AD</b>				
28	Back CTS server	LDAP server	TCP/53 UDP/53	Ensuring DNS name resolution
			UDP/123	Ensuring the operation of the NTP service
			TCP/389 TCP/636	Ensuring LDAP or LDAPS operation
<b>Authentication with ADLDS</b>				
29	Back CTS server	ADLDS server	TCP/389 TCP/636	Ensuring LDAP or LDAPS operation
30	ADLDS server	LDAP server	TCP/389 TCP/636	LDAP or LDAPS user import
<b>Authentication with e-mail</b>				
Connection to the SMTP server is described above (item 7), no additional connections are required				
<b>Authentication with Keycloak</b>				

No.	Source	Destination	Port and protocol	Description
31	External user	Keycloak Front server	TCP/443	User authentication
32	Keycloak Front server	Keycloak Back server	TCP/443	Proxying user requests
33	Internal user	Keycloak Front server	TCP/443	User authentication
34	Keycloak Back server	LDAP server	TCP/389 TCP/636	LDAP or LDAPS user import
35	Back CTS server	Keycloak Back server	TCP/443	Ensuring OpenID operation
	Keycloak Back server	Back CTS server		



## Appendix 3

### ETS, MEDIA AND SINGLE CTS NETWORK INTERACTIONS

No.	Source	Destination	Port and protocol	Description
1	Single CTS server	LDAP server	TCP/389, 636	Ensuring the operation of LDAP or LDAPS
2	Single CTS server Bot server	Bot server Single CTS server	TCP/8000-8100 TCP/443	Interaction of Single CTS with Bot server, interaction of Bot server with Single CTS via the HTTP/HTTPS protocol
3	Internal IS Bot server	Bot server Internal IS	TCP/8000-8100 TCP/80 TCP/443	Interaction of internal information systems with the Bot server, interaction of the Bot server with internal information systems via the HTTP/HTTPS protocol
4	Administrator	Single CTS, ETS, Media, Web Client, XLink servers	TCP/22 TCP/443	Server administration via the SSH protocol eXpress administration via web interface and the HTTPS protocol
5	Single CTS server	SMTP server	TCP/25 TCP/587 TCP/465	Ensuring the sending of e-mails with authentication PIN-code via the SMTP protocol
6	ETS server	Docker registry	TCP/443	Access to the Docker image repository for installing and updating the eXpress software
7	ETS server	DNS and NTP server	TCP/53 UDP/53 UDP/123	Ensuring DNS name resolution Ensuring the operation of the NTP service
8	Internal user	ETS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
9	Single CTS server	ETS server	TCP/5001	Ensuring interaction between the corporate eXpress server and the ETS enterprise server
10	Single CTS server	DNS and NTP server	TCP/53 UDP/53 UDP/123	Ensuring DNS name resolution Ensuring the operation of the NTP service
11	Single CTS, Media, Web Client, XLink servers	Docker registry	TCP/443	Access to the Docker image repository for installing and updating the eXpress software
12	Single CTS	Media	TCP/8188	Conference call management
13	Transcoding	Single CTS server	TCP/443	Transferring transcoded recordings to file storage
14	Transcoding	Media server	TCP/443	Ensuring the transfer of call recording files via HTTPS protocol for subsequent processing by the Transcoding server
15	Internal user	Single CTS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
16	Internal user	Web Client server XLink server	TCP/443 TCP/443	Connecting internal users to the web client Connecting internal users to the XLink server
17	Internal user	Media server	UDP/20000-40000	Ensuring media data transmission via the SRTP conferencing protocol

No.	Source	Destination	Port and protocol	Description
18	ETS server	SMS operator	TCP/443	Sending text messages (SMS) to users
19	ETS server	Huawei push notification service	TCP/443	Sending push notifications to Huawei users
20	ETS server	Apple push notification service	TCP/443	Sending push notifications to iOS users
21	ETS server	Google push notification service	TCP/443	Android push notifications to Android users
22	ETS server	RTS ru.public.express	TCP/5001	Ensuring interaction between the ETS server and the RTS server
23	Let's Encrypt servers	ETS server	TCP/80	Checking the domain for which a certificate is requested from Let's Encrypt
		Single CTS server		
	ETS server	Let's Encrypt servers	TCP/443	Requesting a free certificate from Let's Encrypt
	Single CTS server	Let's Encrypt servers	TCP/443	
24	External user	ETS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
25	External user	Single CTS server (External IP NAT)	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
26	Single CTS server	Partner CTS server	TCP/5001	Ensuring direct message communication between corporate servers bypassing the public contour
	Partner CTS server	Single CTS server (External IP NAT)		
27	Media server	Partner CTS server	UDP/20000-40000	Ensuring media data transmission via the SRTP protocol
	Partner CTS server	Media server (External IP NAT)		
28	External user	Media server (External IP NAT)	TCP/3478	Ensuring the operation of the STUN/TURN protocols
			UDP/3478	
		Media server	UDP/20000-40000	Ensuring media data transmission via the SRTP protocol
29	External user	Web Client server	TCP/443	Client access to the Web Client via the HTTPS protocol
		XLink server	TCP/443	Client access to the XLink server

## Appendix 4

### ETS, MEDIA, FRONT CTS AND BACK CTS SERVER NETWORK INTERACTIONS

No.	Source	Destination	Port and protocol	Description
1	Back CTS server	LDAP server	TCP/53	Ensuring DNS name resolution
			UDP/53	
			UDP/123	Ensuring the operation of the NTP service
			TCP/389	Ensuring the operation of LDAP/LDAPS
			TCP/636	
2	Back CTS server	Bot server	TCP/8000-8100	Back CTS – Bot server interaction via the HTTP/HTTPS protocol
			TCP/443	
	Bot server	Back CTS server	TCP/443	Bot server – Back CTS interaction via the HTTP/HTTPS protocol
3	Internal IS	Bot server	TCP/443	Internal information systems (IS)– Bot server interaction via the HTTP/HTTPS protocol
			TCP/8000-8100	
	Bot server	Internal IS	TCP/80	Bot server – internal IS interaction via the HTTP/HTTPS protocol
			TCP/443	
4	Back CTS server	Docker registry	TCP/443	Access to the Docker image repository to install and upgrade eXpress software
5	Back CTS server	Front CTS server	TCP/443	Monitoring operation of the trusts container and interacting with its API
6	Front CTS server	Back CTS server	TCP/443	Transmission of encrypted user data with TLS transport wrapper
			TCP/2379	Connection to the configuration storage to retrieve various service settings
			TCP/5432	Connection of the trusts container to the database for storing information necessary for operation
			TCP/6379	Connection to Redis
			TCP/9092	Connection to the Kafka software message broker to exchange events between services
7	Administrator	ETS, Front CTS, Back CTS, Media, Web Client, XLink servers	TCP/22	Server administration via the SSH protocol
			TCP/443	eXpress administration via web interface and the HTTPS protocol
8	Back CTS server	SMTP server	TCP/25 TCP/587 TCP/465	Sending emails with authentication PIN-code via the SMTP protocol
9	Back CTS server	Media server	TCP/8188	Conference call management
10	Internal user	Back CTS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
11	Transcoding	Back CTS server	TCP/443	Transferring transcoded recordings to file storage
12	Transcoding	Media server	TCP/443	Ensuring the transfer of call recording files via HTTPS protocol for subsequent processing by the

No.	Source	Destination	Port and protocol	Description
				Transcoding server
13	ETS server	Docker registry	TCP/443	Access to the Docker image repository for installing and updating the eXpress software
14	ETS server	DNS and NTP server	TCP/53	Ensuring DNS name resolution
			UDP/53	
			UDP/123	Ensuring the operation of the NTP service
15	Internal user	ETS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
16	Front CTS server	ETS server	TCP/5001	Ensuring the interaction of the corporate eXpress server with the ETS
17	Front CTS, Media, Web Client server XLink server	Docker registry	TCP/443	Access to the Docker image repository for installing and updating the eXpress software
18	Front CTS server	DNS and NTP server	TCP/53	Ensuring DNS name resolution
			UDP/53	
			UDP/123	Ensuring the operation of the NTP service
19	Internal user	Web Client server	TCP/443	Connecting internal users to the web client
		XLink server	TCP/443	Connecting internal users to the XLink server
20	Internal user	Media server	UDP/20000-40000	Ensuring media data transmission via the SRTP conferencing protocol
21	ETS server	SMS operator	TCP/443	Sending text messages (SMS) to users
22	ETS server	Huawei push notification service	TCP/443	Sending push notifications to Huawei users
23	ETS server	Apple push notification service	TCP/443	Sending push notifications to iOS users
24	ETS server	Google push notification service	TCP/443	Android push notifications to Android users
25	ETS server	RTS ru.public.express	TCP/5001	Ensuring interaction between the ETS server and the RTS server
26	Let's Encrypt server	ETS server	TCP/80	Checking the domain for which a certificate is requested from Let's Encrypt
		Front CTS server		
	ETS server	Let's Encrypt server	TCP/443	Requesting a free certificate from Let's Encrypt
	Front CTS server	Let's Encrypt server	TCP/443	
27	External user	Front CTS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
28	External user	ETS server	TCP/443	Client access to the eXpress corporate contour via the HTTPS protocol
29	Partner CTS server	Front CTS server (External IP NAT)	TCP/5001	Ensuring direct message communication between corporate servers bypassing the public contour
	Front CTS server	Partner CTS server		
30	Partner CTS server	Media server (External IP)	UDP/20000-40000	Ensuring media data transmission via the SRTP conferencing

No.	Source	Destination	Port and protocol	Description
31	External user	NAT)	TCP/3478 UDP/3478 UDP/20000-40000	protocol
		Media server		
		Partner CTS server		
32	External user	Media server (External IP NAT)	TCP/443	Ensuring the operation of the STUN/TURN protocols
		Web Client server		Ensuring media data transmission via the SRTP conferencing protocol
		XLink server		Client access to the Web Client via the HTTPS protocol
			TCP/443	Client access to the XLink server

## Appendix 5

### MONITORING OF EXPRESS OPERATION

The eXpress CS includes third-party software responsible for monitoring the system's operation:

- [Prometheus](#);
- [Grafana](#);
- [Alerts](#).

#### PROMETHEUS

eXpress CS contains a service module (Docker container) with Prometheus monitoring software, which collects metrics from other modules.

Metrics are generated by different modules: `node_exporter`, `cadvisor`, `redis_exporter` and software tools inside the eXpress CS modules.

Prometheus is available at the following path: `/system/prometheus/`. Authentication scheme: basic (encrypted openssl passwd format is supported). Login and password can be found on the Single/Back server in `/opt/express/settings.yaml`.

##### **node\_exporter**

`Node_exporter` exposes OS-level hardware and system metrics provided by \*NIX kernels via metric collectors. `Node_exporter` measures several metrics such as: Memory, Disk, CPU, Network.

If the CTS/ETS servers were deployed using the `dpl` utility, `node_exporter` should be installed automatically.

**Attention!** If the module is installed on a detached server (Front + Back), the installation procedure shall be performed on each server separately.

##### **For installation on a detached server:**

1. Go to the following directory:

```
/opt/express
```

2. Run the following command:

```
dpl nxinstall
ps ax|grep node_exporter | grep -v grep
17802 ? Ssl 322:51 /usr/bin/node_exporter --web.listen-
address=172.17.0.1:9200
```

##### **cAdvisor**

`Cadvisor` is a running daemon that collects, aggregates, processes, and exports information about running containers. In particular, for each container it stores resource isolation parameters, historical resource usage, histograms of total historical resource usage, and network statistics.

If the CTS/ETS servers were deployed using the `dpl` utility, `cAdvisor` should be installed automatically.

If the module is installed on a detached server (Front + Back), the installation procedure shall be performed on each server separately.

## For installation on a detached server:

1. Go to the following directory:

```
/opt/express
```

2. Run the following command:

```
dpl cadvinstall
ps ax|grep cadvisor | grep -v grep
17605 ? Ssl 44:20 /usr/bin/cadvisor -listen_ip 172.17.0.1 -port
9100
```

## Prometheus Federation

If the eXpress SC includes several additional separate servers: a bot server, a separate Media server, or several eXpress SC servers, a Prometheus federation must be deployed for a single monitoring system, which will combine all metrics in one place. The recommended method for centralized collection and storage of metrics from all components of the eXpress SC is as follows:

```
mkdir /opt/prometheus
cd /opt/prometheus
mkdir conf
```

## docker-compose.yaml

Source code:

```
services:
  prometheus:
    image: "prom/prometheus"
    container_name: prometheus
    volumes:
      - "/conf/prometheus.yaml:/etc/prometheus/prometheus.yaml:ro"
      - "prometheus:/prometheus"
    command:
      - '--config.file=/etc/prometheus/prometheus.yaml'
      - '--storage.tsdb.path=/prometheus'
      - '--web.console.libraries=/etc/prometheus/console_libraries'
      - '--web.console.templates=/etc/prometheus/consoles'
      - '--web.route-prefix=/prom/'
      - '--storage.tsdb.retention.time=30d'
    restart: "always"
    security_opt:
      - no-new-privileges

    ports:
      - "8002:9090"

volumes:
  prometheus:
    driver: local
```

Set the relevant parameters (see [Table 59](#)):

[Table 59](#)

Parameter	Description
job_name	Any unique name (e.g., job_name: cts)
basic_auth	Authentication by login/password (for example, basic_auth: <ul style="list-style-type: none"> <li>• username: Prometheus;</li> <li>• password: pass)</li> </ul>
fqnd	the domain name of your CTS\ETS server (e.g.,           static_configs: <ul style="list-style-type: none"> <li>- targets:</li> <li>- 'fqnd:443')</li> </ul>

## conf/prometheus.yaml

Run the following command:

```
docker compose up -d
```

Sample code:

```

scrape_configs:
  - job_name: 'cts' # change, unique field
    scheme: https
    scrape_timeout: 1m
    tls_config:
      insecure_skip_verify: true
    relabel_configs:
      - source_labels: [__address__]
        target_label: federate_host
    basic_auth:
      username: prometheus # change
      password: pass       # change

    honor_labels: true
    metrics_path: '/system/prometheus/federate'

    params:
      'match[]':
        - '{job=~".+"}'

    static_configs:
      - targets:
        - 'fqnd:443' # change CTS FQDN

```

The metrics in built-in Prometheus are stored for 15 days, but if necessary, metrics can be transmitted for long-term storage to a centralized storage compatible with Prometheus (for example, a centralized Prometheus server running in federation mode).

Metrics can be divided into the following groups:

- module status metrics ("on-off", "uptime", "start-up time", etc.);
- performance metrics (CPU usage, memory usage, etc.);
- availability metrics, etc.

Module status metrics are provided in [Table 60](#).

*Table 60*

Components	Module	Metrics
Container status in Docker	Prometheus	up
Postgres database status	Prometheus	pg_up
Redis database status	Prometheus	redis_up

Performance metrics are provided in [Table 61](#).

*Table 61*

Components	Module	Metrics
CPU usage	Zabbix Agent	CPU usage
Memory	Zabbix Agent	Memory usage
Networking	Zabbix Agent	rx/tx rate
SSD	Zabbix Agent	Free space



container: CPU Usage	Prometheus	container_cpu_user_seconds_total
container: Memory Usage	Prometheus	container_memory_usage_bytes
container: SSD	Prometheus	container_fs_writes_bytes_total container_fs_reads_bytes_total
container: Networking	Prometheus	container_network_transmit_bytes_total container_network_receive_bytes_total

Network service availability metrics are provided in [Table 62](#).

*Table 62*

Components	Module	Metrics
Front	Zabbix Server	TCP/80, 443, 3478, 6379, 8188
Front	Zabbix Server	TCP 5001
Back	Zabbix Server	TCP/80, 443, 5432, 9092

Statistical information about the system in provided in [Table 63](#).

*Table 63*

Parameter	Module	Metrics
Registered users	Prometheus	active_users
Users currently connected to the server	Prometheus	online_users
Total number of running Android clients	Prometheus	android_users
Total number of users	Prometheus	total_users
Number of registered users sorted by company name	Prometheus	users_count
Total number of running Web clients	Prometheus	web_users
Total number of messages transmitted	Prometheus	messages_count
Total number of running iOS clients	Prometheus	ios_users
Total number of running Desktop clients	Prometheus	desktop_users
eXpress container versions	Prometheus	express_version
Number of users currently in the call	Prometheus	users_in_calls_count
Postgres database size	Prometheus	pg_database_size
Federated connection status	Prometheus	connection_status

The administrator can manage Janus servers from the administrator web interface, scale the service, and monitor its metrics. Janus server metrics are provided in [Table 64](#).

*Table 64*

Parameter	Module	Metrics
Number of reports of calls rated as unsuccessful by users	Prometheus	call_reports_bad_count
Number of reports of called rated as successful by the users	Prometheus	call_reports_good_count
Number of "I can't hear anyone" reports	Prometheus	call_reports_input_issue_count
Number of "Other" reports	Prometheus	call_reports_other_issue_count
Number of "No one hears me" reports	Prometheus	call_reports_output_issue_count
Number of reports with connection issues	Prometheus	call_reports_poor_connection_count

Parameter	Module	Metrics
Number of reports with screen sharing issues	Prometheus	call_reports_poor_sharing_count
Number of reports with sound issues	Prometheus	call_reports_poor_sound_count
Number of reports with video issues	Prometheus	call_reports_poor_video_count
Number of automatic reports due to unavailability	Prometheus	call_reports_session_issue_count
The number of "disconnected from the call" reports	Prometheus	call_reports_user_disconnected_count
Number of automatic reports due to network disruption	Prometheus	call_reports_webrtc_issue_count
Number of audio	Prometheus	janus_audio_count
Number of users receiving media data	Prometheus	janus_participants_count
Number of participants	Prometheus	janus_publishers_count
Number of recordings	Prometheus	janus_recording_count
Number of rooms	Prometheus	janus_rooms_count
Number of screen sharing sessions	Prometheus	janus_screen_count
Number of videos	Prometheus	janus_video_count
Number of automatic reports due to call business logic errors	Prometheus	redis call_reports_domain_issue_count
Number of calls during the period	Prometheus	voex_call_started_count
Number of conferences during the period	Prometheus	voex_conference_started_count
Number of users who participated in calls/conferences during the period	Prometheus	voex_publisher_joined_count

**To set up**, add appropriate parameters to the settings.yaml file:

**Note.** In case of detached installation, the parameters are added to Back CTS.

```
prometheus_options:
  command:
    - --config.file=/etc/prometheus/prometheus.yml
    - --storage.tsdb.path=/prometheus
    - --storage.tsdb.retention.time=90d
    - --web.console.libraries=/etc/prometheus/console_libraries
    - --web.console.templates=/etc/prometheus/consoles
    - --web.external-url=/system/prometheus
    - --web.route-prefix=/
```

Interface for accessing Prometheus:

- url: specified in the settings.yaml file;
- username: prometheus;
- password: generated in the settings.yaml file during initialization.

## GRAFANA

Grafana is an open source platform for data visualization, monitoring and analysis. Grafana allows users to create dashboards with panels, each displaying specific metrics over a set period of time. Each dashboard is versatile, so it can be customized for a specific project or to suit any development and/or business needs.

Public dashboard for the Single server:  
<https://grafana.com/grafana/dashboards/21386-express-single-cts/>

### To install Grafana with a separate reverse proxy service:

1. On a separate host, create the following directory:
2. In this directory, create a file docker-compose.yaml with the following contents

```
services:
  grafana:
    image: grafana/grafana-enterprise
    container_name: grafana
    environment:
      TZ: Europe/Moscow
    restart: unless-stopped
    volumes:
      - "grafana:/var/lib/grafana"
    ports:
      - "8001:3000"

volumes:
  grafana:
    driver: local
```

3. Run the following command:

```
docker compose up -d
```

After the container is deployed, go to the following address in your browser: <http://ip:8001/>, enter the login/password admin/admin and change the password.

### To set up Grafana:

1. Add a data source. For example, Prometheus. If there are multiple servers, Prometheus federation is recommended.
2. In the Grafana menu, go to Connections > Data sources > Add data source and click "Add data source".
3. Select Prometheus and fill in the form fields (see [Table 65](#)):

Table 65

Parameter	Description
Prometheus server URL	Connect to CTS\ETS servers directly, see <a href="https://fqdn/system/prometheus/">https://fqdn/system/prometheus/</a> . If it is a federation — http(s)://ip/. If you have deployed the federation according to our instructions, and they are in the same docker network — <a href="http://prometheus:9090/prom/">http://prometheus:9090/prom/</a>
Authentication methods	Authentication by login/password (for example, authorization (located on the server in /opt/express/settings.yaml))
Prometheus type	Specify the Prometheus type
Prometheus version	Please specify the version of Prometheus you are using

4. Go to the Dashboards menu and select "Create dashboard".
5. Select "Import dashboard".
6. In the "Find and import dashboards for common applications at grafana.com/dashboards" field, enter "21386" and click "Load".
7. In the CTS-DEMO field, select Prometheus (previously added data-sources) and click "Import".

## ALERTS

Alerts are service notifications. They appear when the system's indicators approach a threshold value or try to go beyond it. eXpress CS monitors the following indicators:

- **System:**
  - CPU;
  - RAM;
  - Hard drive;
  - Drive I/O Utilization.
- **Component interaction:**
  - 5xx errors;
  - 4xx errors;
  - duration of http responses by services.
- **Kafka:**
  - Kafka delays.
- **Docker:**
  - Availability of modules;
  - Availability of trusts;
  - Availability of host;
  - Problems connecting trust service.
- **Postgres:**
  - postgres replication differences exceeded 1 GB;
  - changing the number of Postgres replication nodes.

## SYSTEM

### CPU

Level: **warning**

Recommended trigger response value: **> 80%**

Test duration: **5 minutes**

CPU usage in percent:

```
100 * sum(
  avg(
    rate(node_cpu_seconds_total{mode!="idle"}[10m])
  ) without(cpu)
) without(mode)
```

### RAM

Level: **warning**

Recommended trigger response value: **> 80%**

Test duration: **5 minutes**

RAM usage in percent:

```
(1 - (
  avg_over_time(node_memory_MemAvailable_bytes[10m])
  /
  avg_over_time(node_memory_MemTotal_bytes[10m])
)
```

```
)
) * 100
```

### Hard drive

Level: **warning**

Recommended trigger response value: **> 80%**

Test duration: **5 minutes**

Occupied disk space in percent:

```
(1 - (node_filesystem_avail_bytes{device!~'tmpfs'} /
node_filesystem_size_bytes)) * 100
```

### Drive I/O Utilization

Level: **warning**

Recommended trigger response value: **> 30%**

Test duration: **5 minutes**

Drives utilization in percent:

```
irate(node_disk_io_time_seconds_total{device!~'dm.*'}[5m])
```

## COMPONENT INTERACTION:

### 5xx Errors

This is the total number of 5xx errors when accessing nginx, a rough estimate of the presence of problems (malfunctions — in the case of 4xx errors) in the interaction of components.

Level: **warning**

Recommended trigger response value: **> 20%**

Test duration: **15 minutes**

Percentage of 5xx errors from total number of requests:

```
(
  sum by (express_host) ( avg_over_time(
    rate(http_requests_total{status=~"5.."}[5m])
    [1h:]
  ))
  /
  sum by (express_host) ( avg_over_time(
    rate(http_requests_total[5m])
    [1h:]
  ))
) * 100
```

### 4xx Errors

Level: **warning**

Recommended trigger response value: **> 20%**

Test duration: **15 minutes**

Percentage of 4xx errors from total number of requests:

```
(
  sum by (express_host) ( avg_over_time(
    rate(http_requests_total{status=~"4.."}[5m])
    [1h:]
  ))
  /
  sum by (express_host) ( avg_over_time(
```

```

        rate(http_requests_total[5m])
        [1h:]
    ))
) * 100

```

### Duration of http responses by services

Level: **warning**

Recommended trigger response value: **> 10 seconds**

Test duration: **15 minutes**

### Medium duration of http responses by services:

```

increase(http_request_duration_seconds_sum{app!~'.*_socket'}[5m])
/
increase(http_request_duration_seconds_count[5m])

```

## KAFKA

### Kafka delays

Monitors the speed of components working with data (whether there are problems or not).

Level: **warning, disaster**

Recommended trigger response value: **> 100**

Test duration: **10 min**

Kafka Latencies by topics:

```
sum(kafka_consumergroup_lag) by (topic)
```

## DOCKER

### Availability of modules

The metric collects data on the availability of modules in the system (1 — available, 0 — not available).

Level: **warning, disaster**

Recommended trigger value: **=0**

Test duration: **10 min**

Availability of modules:

```
up
```

### Availability of trusts

State of the trust service (for non-isolated contours or for contours with ETS).

The metric is required to review the status of the service responsible for routing with external contours. Values: 1 — normal state, 0 — module is unavailable, 2+ — fault tolerance error (if any).

Level: **warning, disaster**

Recommended trigger response value: **=0**

Test duration: **5 minutes**

Availability of trusts:

```
up{job="trusts"}
```

### Availability of host

This metric shows the availability of CTS based on the availability of HTTP responses.

Level: **warning, disaster**

*Recommended trigger response value: < 1*

*Test duration: 5 minutes*

Number of http connections:

```
sum by(express_host) (http_connections)
```

### Problems connecting trust service

The metric indicates a routing issue between CTS/eCTS/ETS/RTS.

*Level: warning, disaster*

*Recommended trigger response value: > 0*

*Test duration: 5 minutes*

Trust connections:

```
connection_status{status="red"}
```

---

## POSTGRES

Metrics for systems with postgres failover clusters.

### Postgres replication differences exceed 1 Gb

This metric indicates problems with data replication within the cluster.

*Level: warning, disaster*

*Recommended trigger response value: > 1*

*Test duration: 10 minutes*

Run the following command:

```
sum by(slot_name) (pg_replication_slots_pg_wal_lsn_diff) / 1073741824
```

### Changing the number of Postgres replication nodes

This metric indicates changes in cluster behavior.

*Level: warning, disaster*

*Recommended trigger response value: != <number of cues>*

*Test duration: 10 minutes*

Run the following command:

```
count(pg_replication_slot_slot_is_active)
```

## Appendix 6

### SETTING UP SMARTAPPPROXY HOSTS

If a file from the CTN should become part of a SmartApp Frontend web page (for example, a video in the player), transmission of files through the File Service does not work, since SmartApp Frontend is a separate component.

For this task, there is an option to transmit files via `smartapp_proxy`, which can be used by SmartApp developers.

#### Note.

- this functionality is only available in SmartApp without caching and with proxying;
- these instructions are relevant for CTS server build 3.4 or higher.

#### To set up SmartAppProxy hosts on Single CTS:

1. Add the following lines to the `settings.yaml` file of the CTS server:

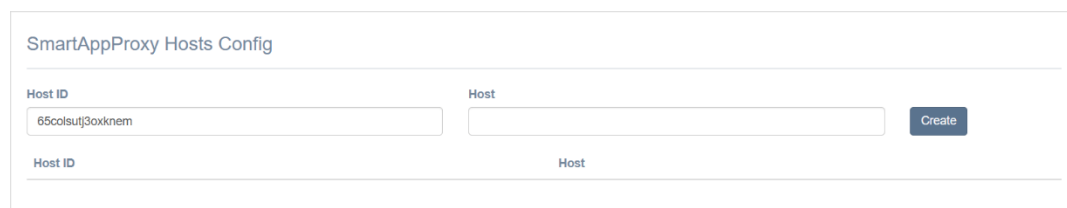
```
smartapp_proxy_enabled: true
smartapp_proxy_env_override:
COOKIE_KEY: _file_service_key
COOKIE_SIGNING_SALT: <salt from file_service or vm5ponDZ built-in default>
```

2. Perform deployment:

```
dpl -p
dpl -d smartapp_proxy admin
```

3. In the SmartApp section of the CTS administrator web interface, in the SmartAppProxy Host Settings block:

- in the "Host ID" field, enter a random set of Latin letters and numbers. This ID is an element of the URL to the proxied file;
- in the "Host" field, enter the URL of the resource from which files will be requested.



SmartAppProxy Hosts Config

Host ID: 65c0lsutj3oxknem

Host:

Create

Figure 45

4. Click "Create".

#### To configure SmartAppProxy hosts on a detached corporate server (Front CTS + Back CTS):

1. Add the following lines to the `settings.yaml` file of the Back CTS server:

```
smartapp_proxy_enabled: true
smartapp_proxy_env_override:
COOKIE_KEY: _file_service key
COOKIE_SIGNING_SALT: <salt from file_service or vm5ponDZ built-in default>
```

2. Perform deployment:

```
dpl -p
dpl -d smartapp_proxy admin
```



3. In the SmartApp section of the CTS administrator web interface, in the SmartAppProxy Host Settings block:
  - in the "Host ID" field, enter a random set of Latin letters and numbers. This ID is an element of the URL to the proxied file;
  - in the "Host" field, enter the URL of the resource from which files will be requested.

SmartAppProxy Hosts Config

Host ID: 65colsutj3oxknem

Host:

Create

Host ID:

Host:

Figure 46

4. Click "Create".

## Appendix 7

### DIAGRAM OF SINGLE CTS NETWORK INTERACTIONS

ATE networking diagram for the deployment of Single CTS is provided below (see [Figure 47](#)).

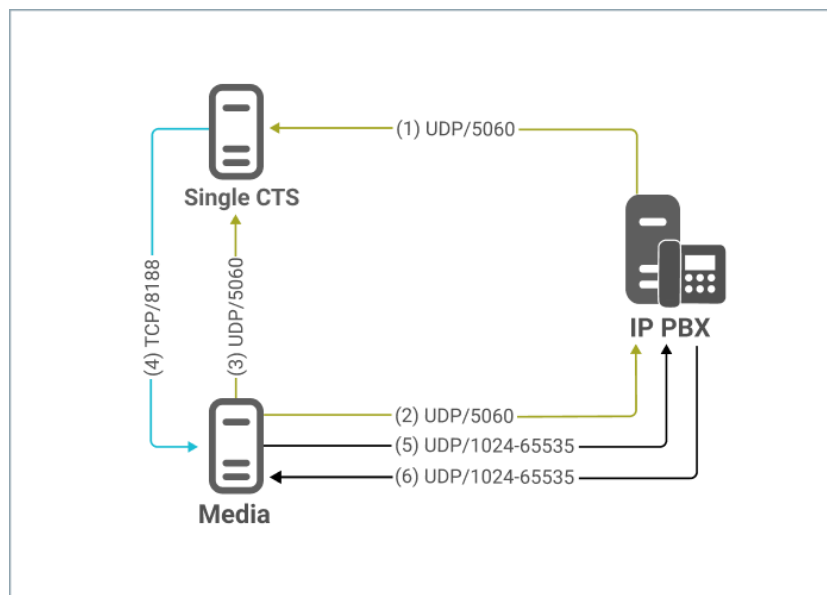


Figure 47. ATE networking diagram for the deployment of Single CTS

Network interactions for the Single CTS deployment scheme are provided in [Table 66](#) (connection numbers in the table correspond to connection numbers in the figure below — see [Figure 47](#)).

Table 66

No.	Source IP	Source port	Destination IP	Destination port	Protocol	Description
1	IP PBX	1024-65535	IP Single CTS	5060	UDP	SIP call signaling from ATE IP
2	IP Media	1024-65535	IP PBX	5060	UDP	SIP call signaling to ATE IP
3	IP Media	1024-65535	IP Single CTS	5060	UDP	SIP call signaling to Single CTS
4	IP Single CTS	1024-65535	IP Media	8188	TCP	Conference server management
5	IP Media	1024-65535	IP PBX	1024-65535	UDP	Media data of call to IP ATE
6	IP PBX	1024-65535	IP Media	1024-65535	UDP	Media data of call to application

## Appendix 8

### ATE NETWORKING DIAGRAM FOR THE DEPLOYMENT OF THE FRONT CTS AND BACK CTS SERVERS

ATE networking diagram for the deployment of the Front CTS + Media and Back CTS servers is provided in the figure below (see [Figure 48](#)).

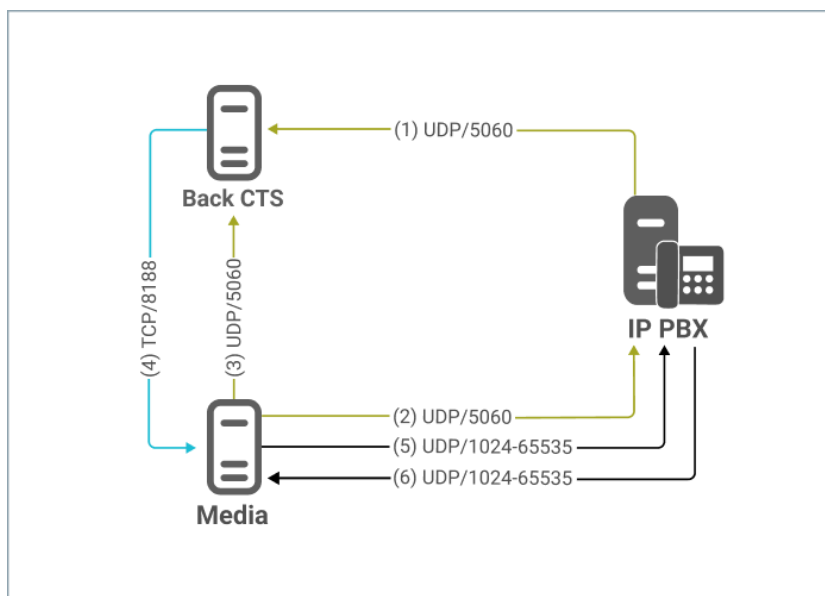


Figure 48. ATE networking diagram for the deployment of Front CTS + Media and Back CTS servers

Network interactions for the Front CTS + Media and Back CTS servers deployment diagram are provided in [Table 67](#) (connection numbers in the table correspond to the numbers in the figure — see [Figure 48](#)).

Table 67

No .	Source IP	Source port	Destination IP	Destination port	Protocol	Description
1	IP PBX	1024-65535	IP Single CTS	5060	UDP	SIP call signaling from ATE IP
2	IP Media	1024-65535	IP PBX	5060	UDP	SIP call signaling to ATE IP
3	IP Media	1024-65535	IP Single CTS	5060	UDP	SIP call signaling to Single CTS
4	IP Single CTS	1024-65535	IP Media	8188	TCP	Conference server management
5	IP Media	1024-65535	IP PBX	1024-65535	UDP	Media data of call to IP ATE
6	IP PBX	1024-65535	IP Media	1024-65535	UDP	Media data of call to application

## Appendix 9

### CTS AND KEYCLOAK INTEGRATION

Keycloak is an open source product for implementing single sign-on. This software allows you to manage identification and access to services and applications. Software license — Apache License 2.0, developed by RedHat, Inc.

Main functions of Keycloak are as follows:

- management of users, groups and roles;
- authentication of client applications using the OpenID Connect and SAML protocols;
- single sign-on;
- support for both relational DBMS and NoSQL (MongoDB);
- clusterisation;
- limited support for OTP authentication (via Google Authenticator);
- integration with external LDAP and Active Directory directories;
- integration with social services (Facebook, Twitter, GitHub, StackExchange etc.);
- expanding functionality through the development of custom SPIs.

### KEYCLOAK REQUIREMENTS

Keycloak version 21.1.2 and above is recommended.

The following conditions must be met during installation:

- HTTPS configuration has been set up;
- Keycloak's public hostname (FQDN) has been specified in accordance with the issued SSL certificate;
- PostgreSQL database is being used.

The following values have been specified in the Realm settings (see [Table 68](#)):

*Table 68*

Parameter	Value	Comment
Tokens -> Access Token Lifespan	8 Hours	Access token lifetime
Sessions -> SSO Session Idle	8 Hours	SSO session timeout
Sessions -> SSO Session	9 Hours	SSO session duration limit

The above intervals are necessary to minimize possible negative consequences during token update operations and to reduce the load on the CTS and Keycloak components.

LDAP federation parameters requirements (if any) (see [Table 69](#)):

*Table 69*

Parameter	Value	Comment
User federation -> LDAP -> Settings -> Import users	ON	Enables user import
User federation -> LDAP -> Settings -> Sync Registrations	ON	New users created by Keycloak will be added to LDAP

Parameter	Value	Comment
User federation -> LDAP -> Settings -> Periodic full sync	ON	Enables periodic full synchronization
User federation -> LDAP -> Settings -> Full sync period	3600	Sets full synchronization period. The value must not be greater than the interval configured in the CTS administrator interface.
User federation -> LDAP -> Settings -> Periodic changed users sync	ON	Periodic synchronization of user changes
User federation -> LDAP -> Settings -> Changed users sync period	300	Changed users synchronization period

Requirements for role transfer (to implement the role model rules): it is necessary to include in the id\_token and userinfo transfer a list of roles available to the Keycloak user.

## REGISTRATION/AUTHORIZATION STEPS

The main steps of user registration/authorization on CTS using Keycloak are shown in the diagram below (see [Figure 49](#)):

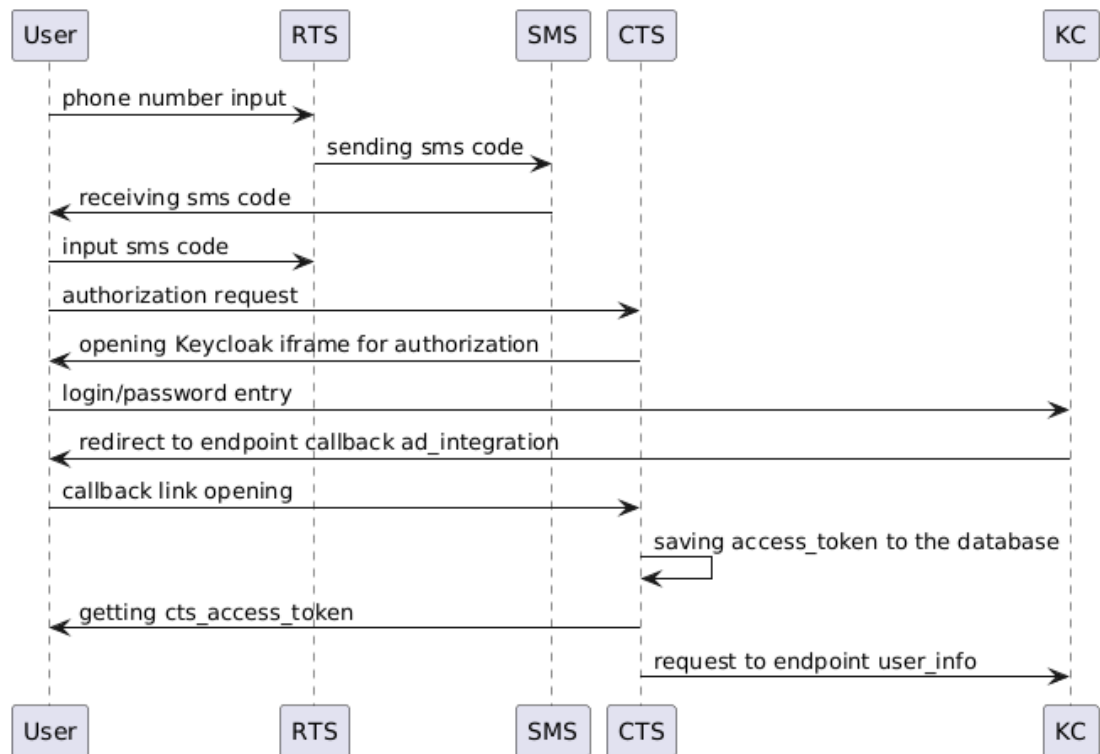


Figure 49. Main steps of user registration/authorization on CTS using Keycloak

## NETWORK INTERACTIONS

There are two options for network interactions.

The user access scheme to the Keycloak interface is shown in the figure below (see [Figure 50](#)):

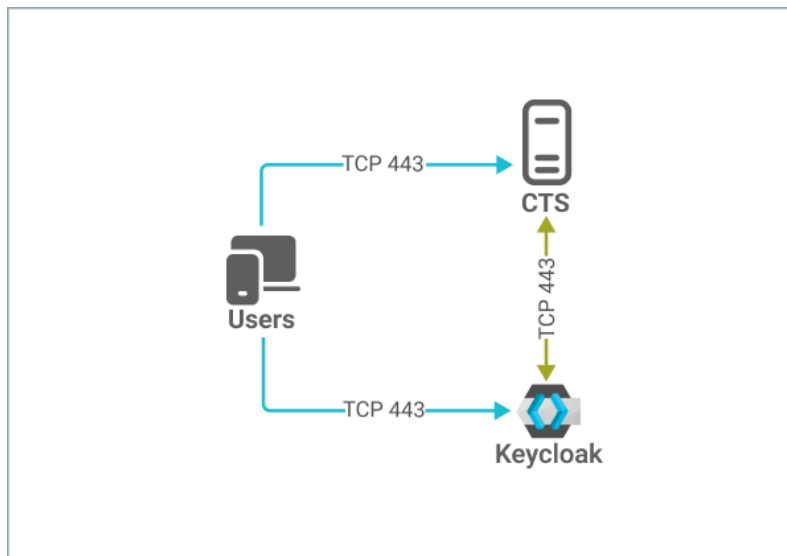


Figure 50. User access to the Keycloak interface

The user access scheme to the Keycloak interface via reverse proxy is shown in the figure below (see Figure 51):

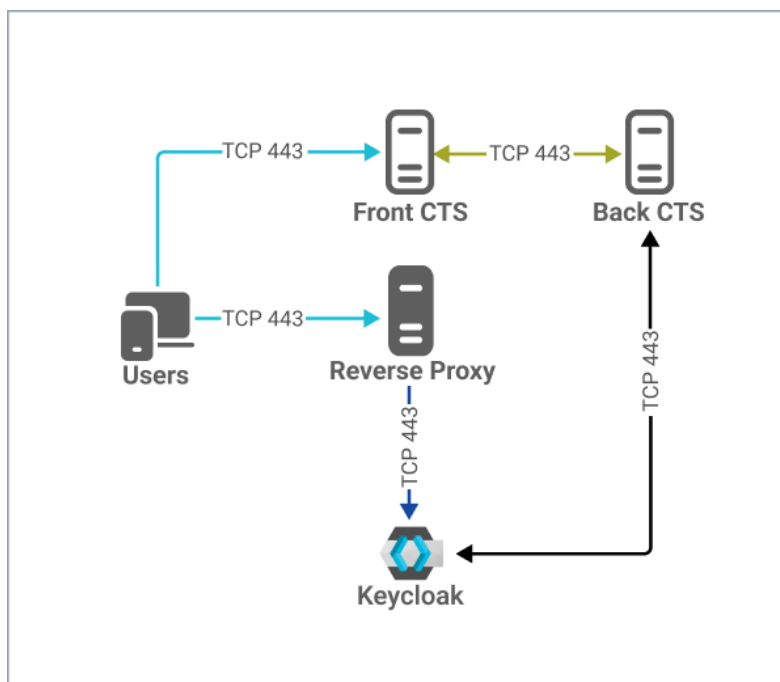


Figure 51. User access to the Keycloak interface via reverse proxy

## SETTING UP INTEGRATION

**Note.** This description of the integration setup is based on the example of the Keycloak administrator console interface version 21.1.2.

The procedure for setting up integration between the CTS server and Keycloak includes the following:

- [Creating client scope](#);
- [setting up field mapping](#);

- [Creating client](#);
- [setting up Keycloak authorization form display](#);
- [setting up QR code authorization](#).

---

## CREATING CLIENT SCOPE

To ensure integration between the CTS server and Keycloak, it is necessary to create a client scope and configure field mapping first:

- `username` — mandatory parameter (in the CTS administrator web interface, specify the correspondence `"Username"` — `"preferred_username"`);
- `user ID` — mandatory parameter;
- `domain` — mandatory parameter (in the CTS administrator web interface, specify the correspondence `"Domain"` — `"domain"`);
- `name` — mandatory parameter;
- `public name` — optional parameter;
- `company` — optional parameter.

Additional mappers are optionally created with the "User Attribute" type and bound to the "user-info" endpoint.

### To create Client Scope:

1. In the Keycloak administrator console, go to the "Client scopes" section.
2. Click the "Create client scope and set the following values (see [Figure 52](#) and [Table 70](#)):

*Table 70*

Parameter	Value
Name	Client scope name. For example, <b>express-scopes</b>
Description	Leave blank
Type	None
Display on consent screen	On
Consent screen text	Leave blank
Include in token scope	On
Display Order	Leave blank

3. Click "Save".

Figure 52

## SETTING UP FIELD MAPPING

### To add field mapping of “User property” type fields:

1. In the created “express-scopes” client scope, select the “Mappers” tab.
2. Click “Configure a new mapper” (see Figure 53).

Figure 53

3. In the “Configure a new mapper” window, select “User Property”.
4. In the window that appears, set the following values (see Figure 54).
  - for the “Username” attribute in accordance with Table 71:

Table 71

Field/switch	Value
Mapper type	User Property
Name	username
Property	username
Token Claim Name	preferred_username
Claim JSON Type	String



Field/switch	Value
Add to ID token	On
Add to access token	On
Add to userinfo	On

- for the "User ID" attribute in accordance with [Table 72](#):

Table 72

Field/switch	Value
Mapper type	User Property
Name	User ID
Property	id
Token Claim Name	user_id
Claim JSON Type	String
Add to ID token	On
Add to access token	On
Add to userinfo	On

- Click "Save".

Figure 54

### To add field mapping of "User attribute" type fields:

- In the created "express-scopes" client scope, select the "Mappers" tab.
- Click "Configure a new mapper" (see [Figure 53](#)).
- In the "Configure a new mapper" window, select "User Attribute".
- In the window that appears, set the following values (see [Figure 55](#)):
  - for the "Domain" attribute (mandatory attribute) in accordance with [Table 73](#):

Table 73

Field/switch	Value
Mapper type	User Attribute
Name	Domain
User Attribute	domain
Token Claim Name	domain

Field/switch	Value
Claim JSON Type	String
Add to ID token	On
Add to access token	Off
Add to userinfo	On
Multivalued	Off
Aggregate attribute values	Off

- for the “Name” attribute (optional attribute) in accordance with [Table 74](#):

*Table 74*

Field/switch	Value
Mapper type	User Attribute
Name	Name
User Attribute:	name
Token Claim Name	name
Claim JSON Type	String
Add to ID token	Off
Add to access token	Off
Add to userinfo	On
Multivalued	Off
Aggregate attribute values	Off

- for the “Public name” attribute (optional attribute) in accordance with [Table 75](#):

*Table 75*

Field/switch	Value
Mapper type	User Attribute
Name	Public name
User Attribute:	public_name
Token Claim Name	public_name
Claim JSON Type	String
Add to ID token	Off
Add to access token	Off
Add to userinfo	On
Multivalued	Off
Aggregate attribute values	Off

- for the “Company” attribute (optional attribute) in accordance with [Table 76](#):

*Table 76*

Field/switch	Value
Mapper type	User Attribute
Name	Company
User Attribute:	company
Token Claim Name	company
Claim JSON Type	String
Add to ID token	Off
Add to access token	Off
Add to userinfo	On
Multivalued	Off
Aggregate attribute values	Off

5. Click "Save".

Client scopes > Client scope details > Mapper details

### Add mapper

If you want more fine-grain control, you can create protocol mapper on this client

Mapper type: User Attribute

Name \* ⓘ

User Attribute ⓘ

Token Claim Name ⓘ

Claim JSON Type ⓘ: String

Add to ID token ⓘ: Off

Add to access token ⓘ: Off

Add to userinfo ⓘ: Off

Multivalued ⓘ: Off

Aggregate attribute values ⓘ: Off

Save Cancel

Figure 55

## CREATING CLIENT

### To create Client:

1. In the Keycloak administrator console, go to the "Clients" section.
2. Click "Create client".
3. In the window that opens, set the following values (see [Figure 56](#)) in accordance with [Table 77](#):

Clients > Create client

### Create client

Clients are applications and services that can request authentication of a user.

1 General Settings

2 Capability config

3 Login settings

Client type ⓘ: OpenID Connect

Client ID \* ⓘ

Name ⓘ

Description ⓘ

Always display in UI ⓘ: Off

Next Back Cancel

Figure 56

Table 77

Parameter	Value
Client type	OpenID Connect
Client ID	Client identification number, for example <b>"express-adintegration"</b>
Name	CTS integration
Description	Leave blank
Always display in UI	Off

- Click "Next".
- In the window that opens, set the following values in accordance with Table 78:

Table 78

Parameter	Value
Client authentication	On
Authorization	Leave blank
Authentication flow	Check the following boxes: <ul style="list-style-type: none"> <li>"Standard flow";</li> <li>"Direct access grants";</li> <li>"Service accounts roles";</li> <li>"OIDC CIBA Grant"</li> </ul>

- Click "Next" and set the following values in accordance with Table 79:

Table 79

Parameter	Value
Root URL	Leave blank
Home URL	Leave blank
Valid redirect URIs	https://cts.company.local/api/v1/ad_integration/openid/success* ("cts.company.local" must be replaced with the address of your CTS/Back CTS server)
Valid post logout redirect URIs	+
Web origins	*

- Click "Save".
- In the Keycloak administrator console, go to the "Clients" section.
- Click "Add client scope".
- Select the previously created "express-scopes" client scope.
- Click the "Add" menu and select "Default".
- Next, in the "Client details" window, set the "offline\_access" scope to "Default".

## SETTING UP KEYCLOAK AUTHORIZATION FORM DISPLAY

### To display Keycloak Authorization Form:

- In the Keycloak administrator console, go to the "Realm settings" section.
- Select the "Security defenses" tab.
- In the "Content-Security-Policy" field, add the following lines:

```
frame-src 'self'; frame-ancestors 'self' https://web.company.local
file:; object-src 'none';
```

**Note.** An example of a web client address is highlighted in red:

- for the CTS server, specify https://corp.express;

- for the ETS server, enter the address of its web client (e.g. <https://web.ets.local>).

4. Click "Save".

---

## SETTING UP QR CODE AUTHORIZATION

**To enable authorization on the CTS server using a QR code**, add the following to the Keycloak server startup command in the command line:

```
--spi-ciba-auth-channel-ciba-http-auth-channel-http-authentication-  
channel-uri=https://ru.public.express  
/api/v1/authentication/openid/ciba/callback
```

**To enable authorization on the ETS server using a QR code**, add the following to the Keycloak server startup command in the command line:

```
--spi-ciba-auth-channel-ciba-http-auth-channel-http-authentication-  
channel-uri=https://ets.corp.lan  
/api/v1/authentication/openid/ciba/callback
```

---

## ROLE MODEL

Within the framework of the role model for individual user groups, the administrator can set restrictions for users with regard to operations with attachments:

- prohibition of sending/forwarding attachments to chats;
- prohibition of downloading/viewing attachments in chats;
- prohibition of the ability to forward/share/save attachments to the device's memory.

Restrictions may apply to:

- attachment type (image, video, document);
- document format (e.g. PDF, DOCX, TXT, etc.);
- attachment size (for example, 300 MB);
- specific chats/channels;
- discussions and chats of calls/conferences;
- users.

First, the administrator creates user groups in the User Groups section to which the restrictions will apply, and then, in the Role Model section, sets the rules that the restrictions will be subject to.

Restrictions can be configured for specific users or specific groups based on server affiliation (for more information, see the document "Administrator's Guide. Volume 2. Operation of the CTS Server").

When creating a group, the administrator can specify the user's OpenID (see [Figure 57](#)).

Create new group

Group name

Enter name

Platform

☐ Android ☐ Desktop ☐ Web ☐ iOS

Connection Type

Select connection type

Add users

Ad groups

Enter AD groups

Openid roles

Enter OpenID roles

Profile fields

Select a position

Select a company

Select a department

Select a domain

Specific users

Enter huid, email, name or ad login

Upload a users list

Excluded users

Specify users who will be excluded from the group

Upload list of excluded users

Create group

Figure 57

For the role model to work correctly, first it is necessary to configure the user role in Keycloak.

### To set up a user role in Keycloak:

1. In the Keycloak administrator interface, in the Client scopes settings, select "roles" (see [Figure 58](#)).

**Client scopes**  
Client scopes are a common set of protocol mappers and roles that are shared between multiple clients. [Learn more](#)

▼ Name 🔍 Search for client scope → [Create client scope](#) Change type to ▼ 1-10 < >

<input type="checkbox"/> Name	Assigned type	Protocol	Display order	Description
<input type="checkbox"/> <a href="#">acr</a>	Default ▼	OpenID Connect	–	OpenID Connect scope for add acr (authentication context class reference) to the token
<input type="checkbox"/> <a href="#">address</a>	Optional ▼	OpenID Connect	–	OpenID Connect built-in scope: address
<input type="checkbox"/> <a href="#">email</a>	Default ▼	OpenID Connect	–	OpenID Connect built-in scope: email
<input type="checkbox"/> <a href="#">express-scopes</a>	None ▼	OpenID Connect	–	–
<input type="checkbox"/> <a href="#">microprofile-jwt</a>	Optional ▼	OpenID Connect	–	Microprofile - JWT built-in scope
<input type="checkbox"/> <a href="#">offline_access</a>	Optional ▼	OpenID Connect	–	OpenID Connect built-in scope: offline_access
<input type="checkbox"/> <a href="#">phone</a>	Optional ▼	OpenID Connect	–	OpenID Connect built-in scope: phone
<input type="checkbox"/> <a href="#">profile</a>	Default ▼	OpenID Connect	–	OpenID Connect built-in scope: profile
<input type="checkbox"/> <a href="#">role_list</a>	Default ▼	SAML	–	SAML role list
<input type="checkbox"/> <a href="#">roles</a>	Default ▼	OpenID Connect	–	OpenID Connect scope for add user roles to the access token

Figure 58

- In the window that opens, go to the “Mappers” tab and select “realm roles” (see Figure 59).

Client scopes > Client scope details

roles [openid-connect](#) Action ▼

Settings Mappers Scope

🔍 Search for mapper → [Add mapper](#) 1-3 < >

Name	Category	Type	Priority
<a href="#">audience resolve</a>	Token mapper	Audience Resolve	30
<a href="#">client roles</a>	Token mapper	User Client Role	40
<a href="#">realm roles</a>	Token mapper	User Realm Role	40

Figure 59

- In the window that opens, activate the options “Add to ID token” and “Add to userinfo” by moving the switch to the right (see Figure 60).

Client scopes > Client scope details > Mapper details

**User Realm Role**  
f34ccd6f-1edb-4f8f-9c26-7528dbfbdf9 Action ▼

Mapper type: User Realm Role

Name \*

Realm Role prefix

Multivalued ☐ On

Token Claim Name

Claim JSON Type

Add to ID token ☐ On

Add to access token ☐ On

Add to userinfo ☐ On

[Save](#) [Cancel](#)

Figure 60

- Check that the settings in the user information are correct (see [Figure 61.](#)).

Settings Keys Credentials Roles **Client scopes** Service accounts roles Sessions Advanced

Setup Evaluate

This page allows you to see all protocol mappers and role scope mappings

Scope parameter   Select scope parameters

Users

```
{
  "sub": "7d2b79ad-34d1-4ba3-a5cf-7cc53e558bc7",
  "email_verified": false,
  "realm_access": {
    "roles": [
      "express-users",
      "offline_access",
      "default-roles-test-03",
      "uma_authorization"
    ]
  },
  "user_id": "7d2b79ad-34d1-4ba3-a5cf-7cc53e558bc7",
  "domain": "express.ms",
  "preferred_username": "testuser1"
}
```

Effective protocol mappers

Effective role scope mappings

Generated access token

Generated ID token

Generated user info

Figure 61.

- Go to OpenID settings in the eXpress administrator web interface (see [Figure 62.](#)).



OpenID Registration Settings

OpenID provider  
KeyCloak < 17 KeyCloak ≥ 17 Bitz

OpenID host  
example: https://openid.provider.com

OpenID port

OpenID realm ID

OpenID client ID

OpenID client secret

OpenID redirect URI

OpenID valid redirect URIs

OpenID response type

OpenID scope

OpenID required role

OpenID role path. Use dot notation for nested path "path.role"  
realm\_access.roles

OpenID async timeout(in ms)  
5000

OpenID login prefill  
Don't prefill

☒ Logout by disabled  
☐ Logout by missing user role

Device authorization method  
CIBA Device Auth Flow

UserID  
user\_id

Public name

Full name

Username

Domain

Company

Position

Department

Avatar

Phone

Telephone Number (Other)

IP Phone

IP Phone (Other)

E-mail

Description

Office

Manager

Other ID

Personnel Number

Business Unit

Personnel Category

Gender

Birthday

Save

Figure 62

6. Specify the required OpenID role – realm-management query-groups.
7. Check the path to roles. Standard path – realm\_access.roles. If it is different, enter the default value.
8. Click "Save".

## CHANGE HISTORY

The “Change History” section contains a list of changes in the document related to changes/modifications of eXpress CS.

### Build 2.5.7

No.	Section	Change	Server	Reference
1.	Setting Up Integration with Active Directory	Requirements for user avatars were added		page <a href="#">91</a>
2.	eXpress Corporate Server Installation	Note corrected		page <a href="#">50</a>
3.	Setting Up Push Notifications	Added a note with the indication of APN Push services	ETS	page <a href="#">74</a>
4.	Appendix 6	Added		page <a href="#">118</a>
5.	Setting Up VoEx Server	The figure have been updated	CTS	page <a href="#">59</a>
6.	Connecting the SMTP Server	Added information in the “E-mail Settings” list	CTS	page <a href="#">88</a>
7.	Setting Up Administrator Authentication	A menu item was renamed in the text	CTS	page <a href="#">89</a>
8.	Terms and Definitions	Added ATE and SIP	CTS	page <a href="#">7</a>
9.	Main Components	Added information about SIP	CTS	page <a href="#">8</a>
10.	Single Corporate Server	Added information about SIP	CTS	page <a href="#">14</a>
11.	Decoupled Corporate Server	Added information about SIP	CTS	page <a href="#">17</a>
12.	Single CTS Installation	Added SIP parameter	CTS	page <a href="#">53</a>
13.	Setting Up VoEx Server	Added	CTS	page <a href="#">59</a>
14.	Setting Up ATE SIP Trunk	Added	CTS	page <a href="#">61</a>
15.	Appendix 7	Added	CTS	page <a href="#">130</a>
16.	Appendix 8	Added	CTS	page <a href="#">130</a>
17.	Appendix 9	Added	CTS	page <a href="#">131</a>

### Build 2.6.0

No.	Section	Change	Server	Reference
1.	Main Components	Added information about the SIP module		page <a href="#">8</a>
2.	Architecture. Single Corporate Server	Added information about ATC connection, architectural diagrams, networking interactions diagrams, removed the ZooKeeper component		page <a href="#">14</a>
3.	Architecture. Decoupled Corporate Server			page <a href="#">17</a>
4.	Appendix 7			page <a href="#">130</a>
5.	Appendix 8			page <a href="#">130</a>
	Architecture. Enterprise Server and Single Corporate Server	Removed the ZooKeeper component		page <a href="#">20</a>
6.	Architecture. Enterprise Server and Decoupled Enterprise Server	Removed the ZooKeeper component		page <a href="#">22</a>

No.	Section	Change	Server	Reference
7.	Single CTS Installation	A parameter for SIP connection was added to the table with available configuration parameters		page <a href="#">54</a>
8.	Setting Up VoEx Server	Added changes to VoEx and SIP server settings	CTS, ETS	page <a href="#">59</a>
9.	Setting Up ATE SIP Trunk	Added section on setting up SIP trunk depending on deployment architecture		page <a href="#">61</a>

## Build 2.7.0

No.	Section	Change	Server	Reference
1.	Architecture	Added explanatory notes about the configuration features, added a description of the Bot server		page <a href="#">12</a>
2.	System requirements	System requirements for the platform were updated		page <a href="#">28</a>
3.	Single Corporate Server	The typical deployment scheme has been updated	CTS	page <a href="#">14</a>
4.	Decoupled Corporate Server	The typical deployment scheme has been updated	CTS	page <a href="#">17</a>
5.	Enterprise Server and Single Corporate Server	The typical deployment scheme has been updated	ETS, CTS	page <a href="#">20</a>
6.	Enterprise Server and Decoupled Enterprise Server	The typical deployment scheme has been updated	ETS, CTS	page <a href="#">22</a>
7.	Appendix 1	The table of network interactions was updated	CTS	page <a href="#">108</a>
8.	Appendix 2	The table of network interactions was updated	CTS	page <a href="#">110</a>
9.	Appendix 3	The table of network interactions was updated	ETS, CTS	page <a href="#">113</a>
10.	Appendix 4	The table of network interactions was updated	ETS, CTS	page <a href="#">115</a>

## Build 2.9.0

No.	Section	Change	Server	Reference
1.	Setting Up Connections for Corporate Servers	Information about filling out the "Name" field was updated	ETS	page <a href="#">84</a>
2.	Web Client Installation	The section was moved	ETS	page <a href="#">46</a>
3.	Setting Up VoEx Server	Section structure was changed		page <a href="#">59</a>
4.	Setting Up Integration with Active Directory	The item dedicated to setting up the visibility of profile fields was updated	CTS	page <a href="#">91</a>
5.	Setting Up SMS Service	Added	ETS	page <a href="#">80</a>

## Build 2.10.0

No.	Section	Change	Server	Reference
1.	DNS Requirements	Added information about the use of Split DNS technology, described the features of its application	CTS	page <a href="#">34</a>

## Build 2.11.0

No.	Section	Change	Server	Reference
1.	Starting up the server	Added a note about creating a server administrator account on Back CTS	CTS	page <a href="#">70</a>

## Build 2.12.0

No.	Section	Change	Server	Reference
1.	Architecture	Added a note about Partner eXpress		page <a href="#">12</a>

## Build 3.0.0

No.	Section	Change	Server	Reference
1.	Architecture	The "logstash" and "elasticsearch" containers were removed from the list of containers		page <a href="#">14</a> page <a href="#">17</a>
2.	Architecture	The "metrics_service" container was added to the list of containers		page <a href="#">14</a> page <a href="#">17</a> page <a href="#">20</a> page <a href="#">22</a>
3.	Eliminating Vulnerabilities	Added an item to the note		
5.	Setting up IP telephony	Added links to Appendices 7 and 8		page <a href="#">61</a>
6.	Update Procedure	Added "OS Update" subsection		page <a href="#">102</a>
7.	Installation Procedure	The procedure for installing corporate servers was updated	CTS	page <a href="#">50</a>
8.	DLP Requirements	DLP requirements were updated		page <a href="#">36</a>
9.	Platform requirements	Platform requirements for the platform were updated		page <a href="#">28</a>
10	Updating Deployka	Throughout the document, the operation DEPLOYKA_SKIP_UPDATE=true was corrected to DPL_PULL_POLICY=never		
11	Network interactions have been updated	Network interactions in the applications have been updated	CTS ETS	page <a href="#">108</a> page <a href="#">110</a> page <a href="#">113</a> page <a href="#">115</a>

## Build 3.1.0

No.	Section	Change	Server	Reference
1.	Platform requirements	Added OS version 22.04 LTS		page <a href="#">28</a>

## Build 3.3.0

No.	Section	Change	Server	Reference
1.	VoEx Server Installation	Updated		page <a href="#">44</a>
2.	Setting Up Registration	Added	CTS	page <a href="#">91</a>

## Build 3.4

No.	Section	Change	Server	Reference
1.	System requirements	System requirements updated with regard to the usage of SSD instead of HDD		page 28

### Build 3.5

No.	Section	Change	Server	Reference
1.	Architecture	Updated		page 12

### Build 3.6

No.	Section	Change	Server	Reference
1.	Architecture	Added "smartapp_proxy" container		page 12
2.	Setting Up SmartAppProxy Hosts	The section has been added		page 128
4.	Chapter 6. Eliminating Vulnerabilities	Deleted		
6.	DLP setup	The command in step 3 has been updated		page 67
7.	VoEx Server Installation	The address in step 11 has been changed.		page 44

### Build 3.7

No.	Section	Change	Server	Reference
1.	Web Client Installation	Updated installation procedure and examples	ETS	page 46
2.	Setting Up Integration with Active Directory	The description of events in Active Directory that will cause an eXpress user to be re-prompted for authentication on the corporate eXpress server has been supplemented	CTS	page 92
3.	Setting Up Push Notifications	A description of connection to Android RuStore has been added	ETS	page 74

### Build 3.8

No.	Section	Change	Server	Reference
1.	Setting Up OpenID	Updated	CTS	page 96
2.	CTS and Keycloak Integration	Created		page 132
4.	Platform requirements	The table "Number of users: 5000" has been added		page 28
5.	Main Components	"For integration with data leak prevention systems that check user messages for prohibited content, the ICAP protocol (TCP/1344 port) is used" has been added.		page 8
6.	Setting Up VoEx Server (STUN and TURN)	Updated		page 59

### Build 3.9

No.	Section	Change	Server	Reference
1.	Decoupled	The Prometheus container has been added to	CTS	page 17

	Corporate Server	the Front CTS server components list		
2.	Starting up the server	Requirements for Administrator password have been added. Checking for errors has been made a separate operation	CTS	page <a href="#">70</a>
3.	Setting Up Integration with Active Directory	The command for OS Ubuntu version 19 and above and other systems in case of error has been added	CTS	page <a href="#">91</a>

### Build 3.10

No.	Section	Change	Server	Reference
1.	Requirements for Storing Videoconferencing Recording Files	Information about the requirements for storing videoconferencing recording files has been added	CTS	page <a href="#">36</a>
2.	Installing Call and Conference Recording Components	Added	CTS	page <a href="#">65</a>
3.	Setting Up Field Mapping	Figure has been added, description of the steps has been supplemented		page <a href="#">136</a>
4.	Update Procedure	The section lists the required updates, removes subsections, and provides a link to a separate document on the update procedure.		page <a href="#">102</a>

### Build 3.11

No.	Section	Change	Server	Reference
1.	Platform requirements	Requirements for operating systems of user PCs have been removed		page <a href="#">28</a>
2.	Installing Call and Conference Recording Components	Requirement to update the CTS server version before installing components has been added, step 2 has been updated	CTS	page <a href="#">65</a>

## Build 3.12

No.	Section	Change	Server	Reference
1.	Single Corporate Server	The name of the Docker container <code>docker_socket_proxy</code> in the list of docker containers of the Single CTS server has been corrected	CTS	page 14
2.	Decoupled Corporate Server	The name of the Docker container <code>docker_socket_proxy</code> in the list of docker containers of the Back CTS server has been corrected	CTS	page 17
3.	Decoupled Corporate Server	Janus has been removed from the list of Back CTS server Docker containers	CTS	page 17
4.	Troubleshooting Typical Errors	Docker container names have been fixed: <code>cts-containername_1</code> ; <code>--tail</code>		page 103
5.	Appendix 1	The table containing information about Single CTS network interactions has been updated. Items 13–19	CTS	page 108
6.	Single Corporate Server	The transcoding, <code>transcoding_manager</code> and <code>recordings_bot</code> containers have been added to the list of CTS server components	CTS	page 14
7.	Decoupled Corporate Server	The transcoding container has been added to Front CTS server components list	CTS	page 17
8.	Decoupled Corporate Server	The transcoding_manager and <code>recordings_bot</code> containers have been added to the list of Back CTS server components	CTS	page 17

## Build 3.13

No.	Section	Change	Server	Reference
1.	Front CTS and Back CTS Servers Installation	The procedure for installing the Front CTS server has been updated (items 6; 7) and the Back CTS server (items 7; 8)	CTS	page 55
2.	Links Server Installation	A section describing the installation of the Links Server has been added	CTS	page 64
3.	Web Client Installation	This section has been moved to Chapter 2 "Installing eXpress"	For all servers	page 46
4.	Installing DLP	A separate section describing the installation of DLP has been added. Previous information on installing DLP has been moved to this section	CTS	page 65

## Build 3.14

No.	Section	Change	Server	Reference
1.	Front CTS and Back CTS Servers Installation	Commands for steps 8 and 9 of Back CTS installation procedure have been added	CTS	page 55
2.	VoEx Server Installation. Pre-Configuration	Updated	CTS	page 48
3.	Installing DLPS on a Dedicated Server	Step 3 of installing a dedicated DLP server has been added	CTS	page 65
4.	VoEx Server Installation	Step 3 title has been updated	CTS	page 48
5.	VoEx Server Installation	Updated	CTS	page 48
6.	Single CTS Installation	Step 3 title has been updated	CTS	page 53
7.	Front CTS and Back	Step 3 title has been updated	CTS	page 55

No.	Section	Change	Server	Reference
	CTS Servers Installation			page <a href="#">56</a>

### Build 3.16

No.	Section	Change	Server	Reference
1.	Name change	The division into volumes has been made. Now the title of the document is "Volume 1. Installation".		
2.	Setting Up Integration with Active Directory	Step 3 of the AD integration setup procedure has been updated	CTS	page <a href="#">91</a>
3.	Platform requirements	Technical requirements for non-fault-tolerant configuration platform have been updated	CTS	page <a href="#">28</a>
4.	Appendix 5. eXpress CTS Monitoring	Table containing information about transmission of Janus metrics to Prometheus has been added	CTS	page <a href="#">118</a>
5.	Appendix 1. Single CTS Network Interactions	The application has been updated	CTS	page <a href="#">108</a>

### Build 3.17

No.	Section	Change	Server	Reference
1.	Architecture	Diagrams of system architecture have been updated. The list of Docker containers has been updated	For all servers	page <a href="#">12</a>
2.	eXpress Monitoring	CTS The table has been updated		page <a href="#">118</a>

### Build 3.18

No.	Section	Change	Server	Reference
1.	DLPS Requirements	The title has been corrected	CTS	page <a href="#">36</a>
2.	Installing DLPS	The title has been corrected	CTS	page <a href="#">65</a>

### Build 3.19

No.	Section	Change	Server	Reference
1.	VoEx Server Installation	The value of the janus_ws_ac parameter in step 9 of the VoEx server installation procedure has been corrected	CTS	page <a href="#">48</a>
2.	Front CTS and Back CTS Servers Installation	The parameter set_real_ip_from in step 8 of the Back CTS server installation procedure has been added	CTS	page <a href="#">56</a>
3.	Appendix 5. eXpress CTS Monitoring	The "Number of users receiving media" parameter has been added to the table containing information about transmission of Janus metrics to Prometheus	CTS	page <a href="#">118</a>
4.	Setting Up Push Notifications	Google FCM address has been added to the note about the necessity of access to APN Push services	ETS	page <a href="#">74</a>



## Build 3.21

No.	Section	Change	Server	Reference
1.	Main Components	A note about the reference nature of the RTS server information in the document has been added	RTS	page <a href="#">8</a>
2.	Regional server	A note about the reference nature of the RTS server information in the document has been added	RTS	page <a href="#">12</a>
3.	Throughout the document	Information about installing and setting up the RTS server has been removed	RTS	

## Build 3.22

No.	Section	Change	Server	Reference
1.	Throughout the document	Due to changes in the system architecture, the Media server is used instead of the VoEx server to provide video and voice communications	CTS	
2.	Architecture	The description of the non-fault-tolerant configuration of eXpress CS for the CTS and the ETS servers has been updated	CTS, ETS	page <a href="#">12</a>
3.	Platform requirements	Requirements for the platform have been updated	CTS	page <a href="#">28</a>
4.	DNS Requirements	Requirements for the DNS have been updated	CTS	page <a href="#">34</a>
5.	Requirements for the Web Client Server	The section has been added	CTS	page <a href="#">36</a>
6.	Requirements for Storing Videoconferencing Recording Files	Information has been updated	CTS	page <a href="#">36</a>
7.	Ubuntu/Debian OS	This section has been updated	CTS	page <a href="#">39</a>
8.	Astra Linux Eagle Operating System	This section has been updated	CTS	page <a href="#">42</a>
9.	Media Server Installation	The section has been reworked, installation parameters have been updated	CTS	page <a href="#">44</a>
10.	Single CTS Installation	The section has been updated, installation parameters have been updated	CTS	page <a href="#">53</a>
11.	Front CTS and Back CTS Servers Installation	The section has been updated, installation parameters have been updated	CTS	page <a href="#">55</a>
12.	Setting Up Media Server	The section has been updated, installation parameters have been updated	CTS	page <a href="#">59</a>
13.	Installing Call and Conference Recording Components	The section has been updated, installation parameters have been updated	CTS	page <a href="#">69</a>
14.	Appendix 2. Front CTS, Media and Back CTS Network Interactions	Information has been updated	CTS	page <a href="#">110</a>
15.	Appendix 3. ETS and Single CTS Network Interactions	Information has been updated	CTS, ETS	page <a href="#">113</a>

No.	Section	Change	Server	Reference
16.	Appendix 4. ETS, Front CTS and Back CTS Network Interactions	Information has been updated	CTS, ETS	page <a href="#">115</a>
17.	Appendix 8. Diagram of Single CTS Network Interactions	Information has been updated	CTS	page <a href="#">130</a>
18.	Appendix 9. ATE networking diagram for the deployment of the Front CTS and Back CTS servers	Information has been updated	CTS	page <a href="#">131</a>

### Build 3.23

No.	Section	Change	Server	Reference
1.	Single CTS Installation	The sample configuration file has been updated	CTS	page <a href="#">54</a>
2.	Setting Up Registration	The registration selection procedure has been updated, the figure has been replaced	CTS	page <a href="#">91</a>
3.	Setting Up E-mail	The figure has been replaced	CTS	page <a href="#">96</a>
4.	Setting Up Open ID	The figure has been replaced	CTS	page <a href="#">96</a>
5.	ATE Networking Diagram for Single CTS	The figure has been replaced	CTS	page <a href="#">130</a>
6.	ATE networking diagram for the deployment of the Front CTS and Back CTS servers	The figure has been replaced	CTS	page <a href="#">131</a>
7.	Setting Up Push Notifications	The table with a description of push notification parameters has been added	ETS	<a href="#">Table 43</a>
8.	Throughout the document	The name of the "Partner eXpress" partner server has been changed to "Partner"	CTS, ETS	

### Build 3.24

No.	Section	Change	Server	Reference
1.	Platform requirements	Technical requirements for the platform have been updated. The note about the need to check the relevance of the versions of additional software being installed has been added	CTS	page <a href="#">32</a>
			ETS	page <a href="#">32</a>
			RTS	page <a href="#">32</a>
2.	Setting Up Registration	The first paragraph has been updated	CTS	page <a href="#">91</a>
3.	Appendix 1	The table containing information about Single CTS network interactions has been updated	CTS	page <a href="#">108</a>
4.	Appendix 2	The table containing information about Front CTS, Media and Back CTS network interactions has been updated	CTS	page <a href="#">110</a>
5.	Appendix 3	The table containing information about ETS, Media and Single CTS network interactions has been updated	CTS	page <a href="#">113</a>
6.	Appendix 4	The table containing information about ETS, Media, Front CTS and Back CTS network	CTS	page <a href="#">115</a>

No.	Section	Change	Server	Reference
		interactions has been updated		

## Build 3.26

No.	Section	Change	Server	Reference
1.	Abstract	The section has been updated		
2.	Throughout the document	"Pre-installed software" has been replaced with "General system software"		
3.	Platform requirements	Added information about the demonstration nature of the software components and a note about the absence of developer liability for the use of demonstration components in a production environment		page <a href="#">33</a>
		"Recordings" replaced with "Transcoding". "AV" replaced with "Media"		<a href="#">Table 3 – Table 16</a>
4.	Single CTS Installation Front CTS and Back CTS Installation Installing DLPS on a Dedicated Server	Added parameters to the configuration file table	CTS	page <a href="#">53</a> page <a href="#">55</a> page <a href="#">65</a>
5.	Web Client Installation	The procedure for installing web client was updated		page <a href="#">46</a>
6.	Setting Up CTS	Added subsection "Setting Up Registration"	CTS	page <a href="#">91</a>
7.	Update Procedure	The section has been updated		page <a href="#">102</a>
8.	Eliminating Vulnerabilities	The section has been added		page <a href="#">106</a>
9.	Additional Capabilities	The section has been added		page <a href="#">118</a>
10.	Setting Up SmartAppProxy Hosts	The host configuration procedure has been updated		page <a href="#">128</a>
11.	Setting Up Keycloak Authorization Form Display	Added a note about web server address	CTS	page <a href="#">140</a>
12.	Throughout the document	The figure have been updated		

## Build 3.27

No.	Section	Change	Server	Reference
1.	Throughout the document	The figure have been updated		Throughout the document
2.	Main Components	The description of the Transcoding server has been added		page <a href="#">8</a>
3.	Architecture	The section has been updated. Diagrams have been replaced, added a description of the Transcoding server		page <a href="#">12</a>
4.	Media Server Installation	The installation procedure has been updated		page <a href="#">48</a> , <a href="#">Table 33</a>
5.	Setting Up Media Server connection to CTS			page <a href="#">59</a>
6.	Transcoding Server Installation			page <a href="#">50</a>
7.	Setting Up JANS, STUN and TURN Servers	The installation procedure has been updated		page <a href="#">59</a>

No.	Section	Change	Server	Reference
8.	Setting Up Push Notifications	The figures and tables have been updated	ETS	page 74, <a href="#">Ошибка! Источник ссылки не найден.</a> , Table 47
9.	Setting Up OpenID	The figure has been updated	CTS	page 96
10.	Appendix 1	The table containing information about Single CTS network interactions has been updated	CTS	page 108
11.	Appendix 2	The table containing information about Front CTS, Media and Back CTS network interactions has been updated	CTS	page 110
12.	Appendix 3	The table containing information about ETS, Media and Single CTS network interactions has been updated	CTS	page 113
13.	Appendix 4	The table containing information about ETS, Media, Front CTS and Back CTS network interactions has been updated	CTS	page 115
14.	Appendix 10	The figures have been updated. Added a description of the role model	CTS	page 132

### Build 3.28

No.	Section	Change	Server	Reference
1.	Decoupled Corporate Server	The figure has been updated	CTS	page 17
2.	Media Server Requirements	Subsection added		page 35
3.	Setting up IP telephony	The section has been updated	CTS	page 61
4.	Connecting the SMTP Server	The section has been updated	ETS	page 73
5.	Setting Up Connections for Corporate Servers	The section has been updated	ETS	page 84
6.	Registration Without Telephone Number	The section has been updated	CTS	page 99
7.	Setting Up Trusted Connections	The section has been updated	CTS	page 99
8.	Appendix 5	The app has been updated		page 118

### Build 3.30

No.	Section	Change	Server	Reference
1.	Media Server Installation	Added transcoding_storage_enabled parameter		page 48
2.	Setting Up SmartAppProxy Hosts	Added information about setting up		page 128
3.	Keycloak Requirements	The section has been added		page 132

### Build 3.31

No.	Section	Change	Server	Reference
1.	Setting Up JANS, STUN and TURN Servers	The information and figure have been updated		page <a href="#">59</a>

### Build 3.32

No.	Section	Change	Server	Reference
1.	Platform requirements	Added information about network bandwidth during videoconferencing		page <a href="#">28</a>
2.	Media Server Requirements	The requirements have been updated		page <a href="#">35</a>
3.	Media Server Installation	Information has been updated		page <a href="#">48</a>
4.	Setting Up OpenID	The field description table has been added, the figure has been updated	CTS	page <a href="#">96</a>
5.	Prometheus	A sample code in conf/prometheus.yaml block has been added		page <a href="#">118</a>
6.	Connecting the TLS Certificate and the Botx SSL Certificate	The figure has been updated	CTS	page <a href="#">86</a>

### Build 3.33

No.	Section	Change	Server	Reference
1.	Setting Up JANS, STUN and TURN Servers	The figure has been updated		page <a href="#">59</a>
2.	Installing DLPS on a Dedicated Server	The note has been updated		page <a href="#">65</a>
3.	Architecture	Figures (diagrams) updated		page <a href="#">12</a>
4.	Types of authentication	The section has been added		page <a href="#">24</a>
5.	Exhibit 1, Exhibit 2, Exhibit 3, Exhibit 4	Tables updated		page <a href="#">108</a> , <a href="#">110</a> , <a href="#">113</a> , <a href="#">115</a>
6.	Setting Up Integration with Active Directory	The table has been updated		page <a href="#">91</a>

### Build 3.34

No.	Section	Change	Server	Reference
1.	Setting Up JANS, STUN and TURN Servers	Figure updated, information on settings added		page 59
2.	Architecture	Figures (diagrams) updated, information on redis on the Media server deleted		page 12
3.	Types of authentication	Figures (diagrams) updated		page 24
4.	Media Server Installation	Information updated (redis replaced by turnserver_shared_key)		page 44
5.	Single CTS Installation			page 53
6.	Front CTS and Back CTS Installation			page 55
7.	Media server connection to the CTS server			page 59
8.	Exhibit 1, Exhibit 2, Exhibit 3, Exhibit 4	Tables updated		page 108, 110, 113, 115
9.	Installation	Server deployment steps updated		page 38
10.	Installing ETS	Information has been updated		page 44
11.	Web Client Installation	Information has been updated		page 46
12.	Setting up IP telephony	Information in the table has been supplemented		page 61

### Build 3.35

No.	Section	Change	Server	Reference
1.	Architecture	Figures (diagrams) updated		page 12
2.	Exhibit 3, Exhibit 4	Tables updated		page 113, 115

### Build 3.36

No.	Section	Change	Server	Reference
1.	Setting Up JANS, STUN and TURN Servers	Information has been updated		page 59
2.	Setting Up Integration with Active Directory	Information has been updated		page 91
3.	Setting Up OpenID	Block about setting levels of access to attributes deleted		page 96