

РУКОВОДСТВО АДМИНИСТРАТОРА

СИСТЕМА «ПЛАТФОРМА НЕЙРОСЕТЬ. Версия 3»

Оглавление

1 Общие положения.....	3
1.1 Полное наименование системы, обозначение.....	3
1.2 Разработчик системы	3
1.3 Назначение документа	4
2 Требования к полигону развертывания	4
2.1 Требования к техническим и программным средствам	4
2.2 Функциональные модули системы.....	5
Core	5
Pool Unit	5
2.3 Требования по обеспечению безопасности	7
3 Порядок установки.....	8

Обозначения и сокращения

Сокращение и термины	Полная форма/определения
ПО	Программное обеспечение
SIP	Протокол передачи данных, описывающий способ установки и завершения пользовательского интернет-сеанса, включающего обмен мультимедийным содержимым
SIP-агент	Подпрограмма, осуществляющая передачу данных по протоколу SIP
Yandex SpeechKit	технология распознавания речи и синтеза речи от компании Яндекс
Ansible	система управления конфигурациями, написанная на языке программирования Python, с использованием декларативного языка разметки для описания конфигураций. Используется для автоматизации настройки и развертывания программного обеспечения.
Playbook (плейбук)	Совместимое с Ansible описание состояния ресурсов системы, в котором она должна находиться в конкретный момент времени, включая установленные пакеты, запущенные службы, созданные файлы и др.

1 Общие положения

1.1 Полное наименование системы, обозначение

Полное наименование системы: **Платформа Нейросеть версия 3**. Эта система позволяет проводить роботизированные кампании по взаимодействию с клиентами через телефонные и текстовые каналы коммуникации, осуществлять настройку деревьев диалога, настройку реплик робота-оператора, планировать и управлять исходящими, входящими обзвонами, настраивать и формировать детальную и сводную отчетность по коммуникациям, слушать записанные аудиозаписи диалогов и читать подробные журналы хода диалога.

Краткое наименование (обозначение) системы: **Платформа НЕЙРОСЕТЬ v3**.

1.2 Разработчик системы

Полное наименование: Общество с ограниченной ответственностью «Нейросеть».

Сокращенное наименование: ООО «Нейросеть».

1.3 Назначение документа

Настоящий документ входит в комплект эксплуатационной документации по Платформе НЕЙРОСЕТЬ v3.

2 Требования к полигону развертывания

2.1 Требования к техническим и программным средствам

Сервер	Кол-во	vCPU	RAM	Disk
mediaservers	10	4	8	130 HDD
Maindb (Postgresql DB)	3	4	4	130 SSD
Pooldb (Postgresql DB)	2	8	8	600 SSD
apicms	2	8	16	130 HDD
glusterfs	3	4	6	200 SSD
nluapi	2	4	8	130 HDD
elasticmasters	3	4	6	60 SSD
elastidata	2	4	6	100 SSD
mongodb	2	6	6	400 SSD
mongodbarbitr	1	2	2	60 SSD
redis	3	6	6	80 SSD
rabbitmq	3	4	4	80 SSD
dnsoptional	2	4	4	60 HDD
haproxy	2	4	4	60 SSD
sbc	2	4	4	60 SSD
etcd	3	6	6	60 SSD
nluengine	2	5	4	130 HDD
Middleware	2	4	8	130 SSD
zabbix	1	2	8	200 HDD

2.2 Функциональные модули системы

Core

Модуль	Описание
<i>Frontend Angular JS</i>	UI системы, написанный на AngularJS.
<i>Backend REST API</i>	REST API для Frontend, а также external API для клиентов, написанная на Python с использованием фреймворка Flask.
<i>File storage GlusterFS</i>	Файловое хранилище на основе распределённой, параллельной, линейно масштабируемой файловой системы GlusterFS. Служит для хранения аудиозаписей фраз, сущностей и т. д.
<i>MongoDB</i>	Служит для хранения статистики диалогов, биллинга и различных логов.
<i>DB SQL Postgres</i>	Операционная SQL БД, служит для хранения пользователей, паролей, прав пользователей, скриптов, фраз и т.д.
<i>RabbitMQ</i>	Содержит в себе загружаемую через API очередь диалогов. Посредством <i>Dialog distributor</i> распределяет диалоги по пулам, в базу PostgreSQL пула.
<i>Redis</i>	Хранит JWT-токены пользователей.
<i>Middleware</i>	Связующее программное обеспечение для реализации нестандартных сценариев клиента. Имеет свою базу данных (<i>DB SQL Postgres</i>).

Pool Unit

Модуль	Описание
<i>PostgreSQL</i>	Хранит в себе операционные данные пула, состояние диалогов, временную статистику по диалогам и биллингу, а также несколько логически реплицируемых таблиц (с Postgres Core'a) с информацией по скриптам и аудиозаписям и т.д.
<i>SBC FreeSWITCH</i>	Session border controller (SBC), используется для сокрытия топологии сети, перекодирования при обработке вызовов на медиа-сервера.
<i>SBC manager</i>	Служит для управления входящими вызовами. Определяет, на какой медиа-сервер можно перевести входящий вызов с <i>SBC FreeSWITCH</i> .

<i>Media server</i>	Управляет медиа-поток, отправляет данные на распознавание, может останавливать звонок, переводить на оператора, делает записи разговоров.
<i>logic_executor_online</i>	Процесс, который обрабатывает онлайн-часть диалогов (звонков).
<i>logic_executor_offline</i>	Процесс, который обрабатывает оффлайн-часть диалогов.
<i>Caller</i>	Запускает звонки (проверка лимитов TCL, временных промежутков, проверка доступности положений).
<i>GlusterFS partition</i>	Является разделом файловой системы сервера с аудиозаписями фраз, сущностей и т. д.
<i>S3_migrator</i>	Скидывает с каждого медиа-сервера записи разговоров в S3 хранилища.
<i>Pool Queue API</i>	Приложение для управления очередью диалогов и звонков. Занимается их приоритизацией.
<i>NLU Engine</i>	Интен-классификатор на основе регулярных выражений. Либо может использовать <i>NLU Model API</i> для классификации на основе нейролингвистической модели.
<i>Message transport</i>	Собирательное понятие из приложений, которые занимаются отправкой и получением сообщений.
<i>ASR/TTS</i>	Сервисы для синтеза и распознавания голоса.

Unit	Модуль	Конфигурационный файл	Лог файл	Управление модулем
Core	Frontend Angular JS	-	-	<i>systemctl status cms.service</i>
	Backend REST API	/etc/api-prod/config.ini	/var/log/api-prod/app.log	<i>systemctl status api.service</i>
	MongoDB	/etc/mongod.conf	/var/log/mongodb/mongod.log	<i>systemctl status mongod.service</i>
	DB SQL Postgres	/opt/PostgreSQL/10/data/postgresql.conf	/opt/PostgreSQL/10/data/log	<i>systemctl status postgresql-10.service</i>
	RabbitMQ	/etc/rabbitmq/rabbitmq.conf	/var/log/rabbitmq/	<i>systemctl status rabbitmq-server.service</i>
	Redis	/etc/redis.conf	/var/log/redis/redis.log	<i>systemctl status redis.service</i>

Pool	PostgreSQL	/opt/PostgreSQL/ 10/data/postgresql.conf	/opt/PostgreSQL/10/data/log	<i>systemctl status postgresql- 10.service</i>
	SBC FreeSWITCH	/etc/freeswitch/ dialplan/internal.xml	/opt/freeswitch/var/log/ freeswitch/freeswitch.log	<i>systemctl status freeswitch</i>
	SBC manager	/etc/sbc-manager/config.ini	/var/log/ivr/sbc-manager.log	<i>systemctl status sbc- manager.service</i>
	Media server	/etc/media-server/config.ini	/var/log/ivr/media-server.log	<i>systemctl status media- server.service</i>
	logic_executor_online	/etc/logic- executor/config.yaml	/var/log/ivr/logic-executor- online.log	<i>systemctl status logic-executor- online.service</i>
	logic_executor_offline	/etc/logic- executor/config.yaml	/var/log/ivr/logic-executor- offline.log	<i>systemctl status logic-executor- offline.service</i>
	Caller	/etc/caller/config.ini	/var/log/ivr/caller.log	<i>systemctl status caller.service</i>
	Pool Queue API	/etc/queue-api/config.ini	/var/log/ivr/queue-api.log	<i>systemctl status queue-api.service</i>
	NLU Engine	/etc/nlu-engine/config.yaml	/var/log/ivr/nlu-engine.log	<i>systemctl status nlu-engine.service</i>
	Message transport	/etc/messaging- api/config.py	-	<i>systemctl status messaging- api.service</i>

2.3 Требования по обеспечению безопасности

Система должна выполнять требования законодательства РФ и подзаконных актов в части защиты персональных данных и тайны связи.

Система должна поддерживать интеграцию с имеющимися системами, и не вводить новые системы, дублирующие функционал имеющихся.

Программный продукт должен иметь механизмы проведения идентификации и аутентификации пользователей.

Система должна обеспечивать уровень протоколирования действий сотрудников и клиентов достаточный для разрешения конфликтных ситуаций. Глубина протоколирования должна определяться действующим законодательством и практикой разрешения споров.

Система поддерживает ролевую модель доступа.

Система поддерживает шифрование данных при взаимодействии с другими приложениями или системами.

3 Порядок установки

Развертывание платформы проходит силами инженеров «Нейросети».

1. По согласованию с заказчиком производят настройку IPSec туннеля.
2. Разворачивают серверы с помощью программного инструмента terraform.
3. Добавляют в inventory для ansible все нужные хосты для установки.
4. Запускают playbook ansible для установки и настройки корневых компонентов из Git:
 - Neuro_repo – развертывание нужных компонентов из своего репозитория, дальнейшее управление и конфигурирование которыми описано выше:
 - PostgreSQL – база данных для хранения операционных данных пула, состояния диалогов, временной статистики по диалогам и биллингу, а также нескольких логически реплицируемых таблиц (с Postgres Core'a) с информацией по скриптам, аудиозаписям и т.д.
 - Media server – сервис для управления медиа-поток, отправляет данные на распознавание, может останавливать звонок, переводить на оператора, делает записи разговоров.
 - Logic_executor_online – процесс, который обрабатывает онлайн-часть диалогов (звонков).
 - Logic_executor_offline – процесс, который обрабатывает оффлайн-часть диалогов.
 - Caller – запускает звонки (проверка лимитов TCL, временных промежутков, проверка доступности положений).
 - Pool Queue API – приложение для управления очередью диалогов и звонков. Занимается их приоритизацией.
 - NLU Engine – интент-классификатор на основе регулярных выражений. Либо может использовать NLU Model API для классификации на основе нейролингвистической модели.
 - Message transport – приложения, которые занимаются отправкой и получением сообщений.
 - ASR/TTS – сервисы для синтеза и распознавания голоса.
 - Python – развертывание высокоуровневого языка программирования.
 - Install_zabbix_agent – развертывание свободной системы мониторинга и отслеживания статусов сервисов, серверов и сетевого оборудования.

- Postgres – разворачивание операционной PostgreSQL для хранения информации о пользователях, аутентификационных данных, правах пользователей, скриптах, фразах и т.д.
 - Etcd – разворачивание быстрой, надёжной и устойчивой к сбоям key-value базы данных.
 - Patroni – разворачивание Python-приложения для создания высокодоступных PostgreSQL кластеров на основе потоковой репликации.
 - Rabbitmq – разворачивание брокера сообщений на основе стандарта AMQP. Содержит в себе загружаемую через API очередь диалогов. Посредством Dialog distributor распределяет диалоги по пулам, в базу PostgreSQL пула.
 - Elasticsearch – разворачивание программной системы для полнотекстового поиска.
 - Redis – разворачивание NoSQL СУБД, работающей со структурами данных типа «ключ — значение». Хранит JWT-токены пользователей.
 - Mongodb – разворачивание документно-ориентированной СУБД, не требующей описания схемы таблиц. Служит для хранения статистики диалогов, биллинга и различных логов.
 - Freeswitch – разворачивание Session border controller (SBC), используется для сокрытия топологии сети, перекодирования при обработке вызовов на медиа-сервера.
 - Gfs_server – разворачивание файлового хранилища на основе распределённой, параллельной, линейно масштабируемой файловой системы GlusterFS. Служит для хранения аудиозаписей фраз, сущностей и т.д.
 - Gfs_clients – разворачивание разделов файловой системы сервера с аудиозаписями фраз, сущностей и т.д.
 - Нарошу – разворачивание ПО для обеспечения высокой доступности и балансировки нагрузки для TCP и HTTP-приложений, посредством распределения входящих запросов на несколько обслуживающих серверов.
 - Keepalived – разворачивание процесса, позволяющего организовать отказоустойчивость сервиса и балансировку нагрузки.
 - Dns (опционально) – разворачивание распределённой системы для получения информации о доменах.
5. Производят настройку структуры MongoDB с помощью уже имеющихся конфигурационных файлов в Git.
 6. Производят настройку компонентов платформы так же по уже имеющимся конфигурационным файлам в Git.
 7. Запускают все компоненты с помощью ci в Git.
 8. Производят настройку SIP-интеграций и интеграций с системами заказчика (Billing, CRM и т.д.)

9. Производят настройку системы мониторинга Zabbix с помощью playbook ansible.
10. Проводят интеграционное тестирование системы.
11. Передают систему в эксплуатацию.