


v1.26

1.26


Eltex SC - OS Ubuntu 20.


- — 1;
- — i5 3,0 ;
- — 8 ;
- — 1000 ;
- (/) — 2000 IOPS.


 .

Eltex SC


Eltex SC 1.26 Ubuntu 20.04. Eltex SC (1.25) : [Eltex SC](#).

 1.19.4 deb- .

 1.25 eltex-notification-gw(ngw) **notification.properties.j2.** **vars/default.yml**

 Eltex SC , , !

Ansible . Ansible .


 Ansible — , Python . , Eltex SC.

:

1. Ansible Ubuntu 20.04.

Ansible

```
apt update
apt install --install-recommends linux-generic-hwe-20.04-edge
apt install software-properties-common
add-apt-repository --yes --update ppa:ansible/ansible
apt install ansible
```

 Ansible .

2. (v2.9):

| Ansible |
|-------------------|
| ansible --version |

3. Ansible .


```
_:
ansible-galaxy collection install community.general
ansible-galaxy collection install community.crypto
ansible-galaxy collection install community.docker
```

4. .

```
tar.gz, /etc root.
_:
tar -C /etc -xvf ansible-iot-1.26.tar.gz
```

(/) Ansible /etc/ansible-iot-1.26

5. /etc/ansible-iot-1.26/inventory
, nano. root ansible_sudo_pass:

 root rootpasswd

```
_:
inventory

[IoT]
localhost ansible_connection=local ansible_sudo_pass=rootpasswd
[elk]
localhost ansible_connection=local ansible_sudo_pass=rootpasswd
[monitoring]
localhost ansible_connection=local ansible_sudo_pass=rootpasswd
```

6. Eltex SC.

/etc/ansible-iot-1.26/vars/default.yml

, nano. IP- server_name:

vars/default.yml

```
---
#
iot:
  # (IP-) ,            IoT
  # 'localhost',
  # !!! 'serverName'        (IP-),
  # 'localhost',            'localhost'
  serverName: "my.test.server"
  #
  installDir: /storage/iot

#        (Elasticsearch + Logstash + Kibana)
elk:
  #        appender,        logstash
  #        ,    ELK        ;
  #
  enable: false
  # (IP-) ,        ELK
  #        'iot.serverName',        (    )
  #        [iot] [monitoring]
  serverName: "{{ iot.serverName }}"
  #
  installDir: /storage/elk

#        (Prometheus + Grafana)
monitoring:
  # (IP-) ,            (Prometheus + Grafana)
  #        'iot.serverName',        (    )
  #        [iot] [elk]
  serverName: "{{ iot.serverName }}"
  #
  installDir: /storage/monitoring
```

vars/default.yml

```
# MongoDB
mongodb:
  # MongoDB. , , `4`
  version: 5
  external:
    # true, MongoDB
    # !!! MongoDB , 'addr' 'port'
    enable: false
    # MongoDB
    addr: "{{ iot.serverName }}"
    # MongoDB
    port: 27017

# WEB
web:
  # (IP-) , WEB
  # 'iot.serverName', ( )
  serverName: "{{ iot.serverName }}"
  # HTTP, WEB
  httpPort: 80
  # HTTPS, WEB
  httpsPort: 443
  # HTTP HTTPS
  redirectHttpToHttps: false
  certbot:
    # certbot Let's Encrypt
    enable: false
    # E-mail . Let's Encrypt
    email: test@email.com

# e-mail (NGW)
mail:
  smtp:
    submitter: test@email.com
    password: "password"
    senderPrefix: " Eltex-SC"
    auth: "true"
    host: email.com
    port: 587

# IoT core
core:
  # IoT Core
  logLevel: INFO

  # Z-Way
  ctlGate:
    port: 8070
    tcpPort: 8069
    sslPort: 8072
```

vars/default.yml

```
# API
api:
  port: 8071
  sslPort: 8073

# CAPTCHA: easy, medium, hard
captchaLevel: "easy"

server:
  # HTTPS WEB ('true' , ,
  # 'web.httpsPort'). 'false', HTTP , 'web.httpPort'
  useHttpsForUi: true
  # HTTPS (, )
  useHttpsForApi: false
  # HTTPS
  useHttpsForCameraLinks: true
  # 'web.serverName' 'iot.serverName' 'web.httpPort'/'web.httpsPort'
  # 'core.api.port'/'core.api.sslPort' API
  useUiProxyForApi: false

#
selfRegistration:
  allow: true
  allowDemo: true

#
video:
  # Flussonic
  flussonic:
    url: ""
    apiKey: ""
    operatorId: ""
    adminLogin: ""
  # eltex
  eltex_server:
    url: ""

# ().
yandexSkill:
  # Basic Authentication
  clientId: "YandexClientIdChangeMe"
  password: "PasswordChangeMe"
  # ID ,
  skillId: ""
  # OAuth-,
  oauthToken: ""
```

vars/default.yml

```
#
sberSkill:
  # Basic Authentication
  clientId: "SberClientIdChangeMe"
  password: "PasswordChangeMe"
  # Bearer-,
  bearerToken: ""

# Mail.ru . /
marusyaSkill:
  # Basic Authentication
  clientId: "MarusyaClientIdChangeMe"
  password: "PasswordChangeMe"
  # App ID, VK
  appId: "MarusyaAppIdChangeMe"
  # OAuth-,
  oauthToken: ""
```



, , e-mail. :

mail:

smtp:

submitter — e-mail;

password — e-mail;

auth — smtp ();

senderPrefix — ;

host — smtp-;

port — smtp- .

vars/service_parameters.yml :

/vars/service_parameters.yml

```
---
#
release: 1.26

# docker registry, docker-
registry: hub.eltex-co.ru

# ( restart_*.yaml)
# ,
```

/vars/service_parameters.yml

```
services:

#      .      ""
#      .      IoT  Ansible,
#
withDistroPreparingStep: true

# ,      (      )
containerNameSuffix: ""

# ,      docker (      docker)
networkNameSuffix: ""

# IoT ( docker-compose),
# 'enable' -      docker-compose.yml
# 'port.map' -
# 'port.export' -
# 'db.name' - ,      ( )
iotServices:
  db:
    port:
      map: 27017
      export: false
  broker:
    enable: true
    external:
      port:
        map: 8883
    internal:
      port:
        map: 8083
        export: false
  db:
    name: iot-broker
olapService:
  enable: true
  port:
    map: 8023
    export: false
  db:
    name: iotcore
    port:
      map: 8123
      export: false
```

/vars/service_parameters.yml

```
ngw:
  port:
    map: 8040
    export: false
  db:
    name: notification-gw
    user: javauser
    password: javapassword
    port:
      map: 3306
      export: false
captcha:
  port:
    map: 8088
    export: false
  caseSensitive: true
  allowedSizes:
    - "312x45"
    - "270x40"
zscaptcha:
  port:
    map: 8089
    export: false
  caseSensitive: true
core:
  # , core
  enable: true
web:
  # WEB, web
  enable: true

# ELK ( docker-compose ),
# 'port.map' -
# 'port.export' -
elkServices:
  elasticsearch:
    rest:
      port:
        map: 9200
    nodes:
      port:
        map: 9300
  logstash:
    port:
      map: 5001
    api:
      port:
        map: 9600
  kibana:
    port:
      map: 5601
```


/vars/service_parameters.yml

```
# ( docker-compose ),
# 'port.map' -
# 'port.export' -
monitoringServices:
  prometheus:
    port:
      map: 9090
  grafana:
    port:
      map: 3000
  nginxExporter:
    enable: true
    port:
      map: 9113

coreInternal:
  #
  testdata:
    enable: false
  # swagger ( API )
  swagger:
    enable: false
  # MQTT Broker
  mqttbroker:
    enable: true
  # OlapService
  olapservice:
    enable: true
  # MongoDB,
  core:
    db:
      name: iot-core
  fs:
    db:
      name: iot-fs
  licenses:
    db:
      name: iot-licenses
  events:
    db:
      name: iot-events
  mjollnir:
    # URL Mjollnir
    url: "http://smart.eltex-co.ru:8078/api/v1"
```

7. :

```
cd /etc/ansible-iot-1.26
ansible-playbook install_iot.yml
```



Eltex SC c MongoDB, mongodb.org APT (, </etc/apt/sources.list.d/mongodb-org-4.4.list>).

8. :

```
docker ps
```

| docker ps | | | | | | |
|--------------|--|-------------------------|---------------|--------------|--|---------------------------|
| CONTAINER ID | IMAGE | COMMAND | CREATED | STATUS | PORTS | NAMES |
| 25c08d17a4ae | hub.eltex-co.ru/iot-clickhouse-backup:1.26 | /entrypoint.sh /ent... | 2 minutes ago | Up 2 minutes | | iot-olapservice-db-backup |
| 36c21b863cc9 | hub.eltex-co.ru/iot-double-web:1.26 | /docker-entrypoint.... | 2 minutes ago | Up 2 minutes | 0.0.0.0:80->80/tcp, :::80->80/tcp, 0.0.0.0:443->443/tcp, :::443->443/tcp | iot-double-web |
| 01fe2697e5ff | hub.eltex-co.ru/iot-core:1.26 | java -Dspring.profi... | 2 minutes ago | Up 2 minutes | .0.0.0:8069-8073->8069-8073/tcp, :::8069-8073->8069-8073/tcp | iot-core |
| 07d1f93831bd | hub.eltex-co.ru/iot-mqttbroker-mongo:1.26 | java -cp @/app /jib-... | 2 minutes ago | Up 2 minutes | 0.0.0.0:8883->8883/tcp, :::8883->8883/tcp | iot-mqtt-broker |
| d1c736dc27d0 | hub.eltex-co.ru/eltex-ngw:1.26-602 | /usr/sbin /ngw_start... | 2 minutes ago | Up 2 minutes | | iot-ngw-sc |
| 228d41c96cba | hub.eltex-co.ru/iot-olapservice:1.26 | java -cp @/app /jib-... | 2 minutes ago | Up 2 minutes | | iot-olapservice |
| e8e2899f2c8d | hub.eltex-co.ru/iot-captcha:1.26 | java -jar LibreCapt... | 2 minutes ago | Up 2 minutes | 8888/tcp | iot-captcha |
| 57c02941cc4f | hub.eltex-co.ru/iot-mongo5:1.26 | /entrypoint.sh | 2 minutes ago | Up 2 minutes | 0.0.0.0:27017->27017/tcp, :::27017->27017/tcp | iot-mongo |
| 7c3d8d5c4137 | hub.eltex-co.ru/iot-mysql:1.26 | docker-entrypoint.s... | 2 minutes ago | Up 2 minutes | 3306/tcp, 33060/tcp | iot-iot-mysql |
| 9009dd4cd675 | hub.eltex-co.ru/iot-olapservice:1.26 | java -cp @/app /jib-... | 2 minutes ago | Up 2 minutes | 0.0.0.0:8023->8023/tcp, :::8023->8023/tcp | iot-olapservice |
| 6c61b34c3a41 | hub.eltex-co.ru/iot-clickhouse-server:1.26 | /entrypoint.sh | 2 minutes ago | Up 2 minutes | 9000/tcp, 0.0.0.0:8123->8123/tcp, :::8123->8123/tcp, 9009/tcp | iot-olapservice-db |

```
Ⓢ : http://[ Eltex SC]
server_name /etc/ansible-iot-1.26/vars/default.yml
```

API .

9. .

 , .

Eltex SC MongoDB

MongoDB, .. Eltex SC / mongodb:

 mongodb.org . deb- (,).

1. MongoDB (4.4):

```
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install -y software-properties-common gnupg build-essential net-tools dkms
wget https://www.mongodb.org/static/pgp/server-4.4.asc
sudo apt-key add server-4.4.asc
echo "deb [ arch=amd64,arm64 ] https://repo.mongodb.org/apt/ubuntu focal/mongodb-org/4.4 multiverse" | sudo tee
/etc/apt/sources.list.d/mongodb-org-4.4.list
sudo apt-get update
sudo apt-get install -y mongodb-org
sudo service mongod start
sudo service mongod status
```

2. `/etc/mongod.conf` net :

```
port: 27017
bindIp: 0.0.0.0
```

mongod:

```
sudo service mongod restart
sudo service mongod status
```

3. Ansible :

```
sudo add-apt-repository --yes --update ppa:ansible/ansible
sudo apt install ansible
sudo ansible-galaxy collection install community.general
sudo ansible-galaxy collection install community.crypto
sudo ansible-galaxy collection install community.docker
```

4. :

```
tar -C /etc -xvf ansible-iot-1.26.tar.gz
```

. (/) Ansible `/etc/ansible-iot-1.26`.

5. `/etc/ansible-iot-1.26/inventory`.

, nano. root ansible_sudo_pass:



root rootpasswd

inventory

```
[iot]
localhost  ansible_connection=local  ansible_sudo_pass=rootpasswd
[elk]
localhost  ansible_connection=local  ansible_sudo_pass=rootpasswd
[monitoring]
localhost  ansible_connection=local  ansible_sudo_pass=rootpasswd
```

6. , . `/etc/ansible-iot-1.26/vars/default.yml` `/etc/ansible-iot-1.26/vars/service_parameters.yml`.

default.yml , nano. IP- **server_name**.

enable true, 'addr' 'port'.

vars/default.yml

```
mongodb:
# MongoDB. , , `4`
version: 5
external:
# true, MongoDB
# !!! MongoDB , 'addr' 'port'
enable: false
# MongoDB
addr: "{{ iot.serverName }}"
# MongoDB
port: 27017
```

8. :

```
cd /etc/ansible-iot-1.26
sudo ansible-playbook install_iot.yml
```



: http://[Eltex SC]

server_name `/etc/ansible-iot-1.26/vars/default.yml`

9. .



, .

:

```
ansible-playbook restart_iot.yml --extra-vars '{"services":["web", "core", "broker", "olapservice"]}'
```

```
, ( "web", "core", "broker", "olapservice"). "services" --extra-vars, .
:
```

```
ansible-playbook stop_iot.yml --extra-vars '{"services":["web", "core", "broker", "olapservice"]}'
```

```
:
```

```
ansible-playbook update_iot.yml
```

```
:
```

```
ansible-playbook install_iot.yml
```

**/etc/ansible-iot-1.26/templates : /etc/ansible-iot-1.26/templates/iot/default-for-docker.yml.j2 - : /etc/ansible-iot-1.26/templates/iot/web
/base_config**

: /storage/iot/core/var/log/eltex-sc/server.log

. : server-YYYY-MM-DD.NN.log, YYYY-MM-DD — , NN — .



install_iot.yml:

ansible-playbook install_iot.yml

/etc/ansible-iot-1.26/vars/default.yml . , /etc/ansible-iot-1.26/vars/default.yml, , .

default-for-docker.yml.j2:

```

#      : , , ,
testData:
  environment: {{ 'true' if coreInternal.testdata.enable else 'false' }}

fileStorage:
  path: "/var/lib/eltex-sc/files"

controllerGateTCP:
  port: {{ core.ctlGate.tcpPort }}

controllerGate:
  port: {{ core.ctlGate.port }}

controllerGateSecurity:
  port: {{ core.ctlGate.sslPort }}
  key: "/etc/ssl/private/eltex-sc-ctl-gate.key"
  crt: "/etc/ssl/certs/eltex-sc-ctl-gate.crt"

api:
  port: {{ core.api.port }}
  sslPort: {{ core.api.sslPort }}

ui:
  serverName: "{{ web.serverName }}"
{% if web.httpPort != 80 or web.httpsPort != 443 %}
# UI      ,      UI
# .      port/sslPort      URL  UI (WEB)
{% if web.httpPort != 80 %}
  port: {{ web.httpPort }}
{% endif %}
{% if web.httpsPort != 443 %}
  sslPort: {{ web.httpsPort }}
{% endif %}
{% endif %}

```

```

mqttBroker:
  enabled: {{ 'true' if coreInternal.mqttbroker.enable else 'false' }}
  # MQTT-,
  host: "broker"
  # MQTT-,      MQTT
  port: 8883
  # ,      REST API,
  apiPort: 8083
  # URL,      MQTT-, MQTT-
  remoteAccessURL: "{{ iot.serverName }}:{{ iotServices.broker.external.port.map }}"
  # ,      "Offline" MQTT-
  offlineTimeoutSec: 300

olapservice:
  enabled: {{ 'true' if coreInternal.olapservice.enable else 'false' }}
  host: "olapservice"
  port: 8023

captcha:
  # CAPTCHA (Libre CAPTCHA, Zero Storage Captcha)
  instances: captcha:8088, zs-captcha:8089
  proportions: 30, 70
  level: "{{ core.captchaLevel }}"

resilience4j:
  # ,
  #
  circuitbreaker:
    configs:
      default:
        # ,
        failureRateThreshold: 50
        # ,
        slowCallRateThreshold: 50
        # ,
        slowCallDurationThreshold: 5s
        # ,
        permittedNumberOfCallsInHalfOpenState: 5
        # ,      (COUNT_BASED/TIME_BASED)
        slidingWindowType: COUNT_BASED
        #
        slidingWindowSize: 100
        #
        automaticTransitionFromOpenToHalfOpenEnabled: true
        # ,      circuitbreaker
        waitDurationInOpenState: 5s

```

```

#
  minimumNumberOfCalls: 20
instances:
  eventlog:
    baseConfig: default
  flussonic:
    baseConfig: default
    slowCallDurationThreshold: 3s
    waitDurationInOpenState: 20s
  ivideon:
    baseConfig: default
    slowCallDurationThreshold: 3s
    waitDurationInOpenState: 20s
#
timelimiter:
  configs:
    default:
      # ,
      timeoutDuration: 10s
  instances:
    eventlog:
      baseConfig: default
    flussonic:
      baseConfig: default
      timeoutDuration: 3s
    ivideon:
      baseConfig: default
      timeoutDuration: 3s
#
bulkhead:
  configs:
    default:
      # , bulkhead
      maxConcurrentCalls: 20
  instances:
    eventlog:
      baseConfig: default
    flussonic:
      baseConfig: default
      maxConcurrentCalls: 10
    ivideon:
      baseConfig: default
      maxConcurrentCalls: 10

```



```

# . : ,
retry:
  configs:
    default:
      # ( )
      maxAttempts: 2
      # ( )
      waitDuration: 500
  instances:
    eventlog:
      baseConfig: default
    flussonic:
      baseConfig: default
    ivideon:
      baseConfig: default
# .
ratelimiter:
  configs:
    default:
      # , ( )
      timeoutDuration: 25ms
      # ( )
      limitRefreshPeriod: 1000
      # ,
      limitForPeriod: 10
  instances:
    eventlog:
      baseConfig: default
    flussonic:
      baseConfig: default
    ivideon:
      baseConfig: default
# UI(WEB) API
server:
  # ,
  name: "{{ iot.serverName }}"
  # https:// UI
  useHttpsForUi: {{ 'true' if core.server.useHttpsForUi else 'false' }}
  # https:// API
  useHttpsForApi: {{ 'true' if core.server.useHttpsForApi else 'false' }}
  # https://
  useHttpsForCameraLinks: {{ 'true' if core.server.useHttpsForCameraLinks else 'false' }}
  # ui.serverName server.name ui.port/ui.sslPort api.port/api.sslPort
  # API
  useUiProxyForApi: {{ 'true' if core.server.useUiProxyForApi else 'false' }}
  oauth2:
    # access- . 5
    # , authorization server 60 ,
    # . : accessTokenTimeToLive + 60
    accessTokenTimeToLive: 3600

```

```

# jetty
jetty:
  #
  connection-idle-timeout: 120000ms
  max-http-form-post-size: 200000B
  #
  threads:
    acceptors: -1
    selectors: -1
    #
    idle-timeout: 120000ms
    #
    min: 32
    #
    max: 256
    #
    max-queue-capacity: 32768

selfRegistration:
  allow: {{ 'true' if core.selfRegistration.allow else 'false' }}
  allowDemo: {{ 'true' if core.selfRegistration.allowDemo else 'false' }}

electricMeterScheduler:
  cron: "0 0/30 * * * ?"

electricMeterArchiveScheduler:
  cron: "0 0 17 * * ?"

batch:
  maxPoolSize: 2
  corePoolSize: 1

services:
  ngw:
    host: "ngw-sc"
    port: 8040
  alarmService:
    enabled: false
  loginInfo:
    # loginInfo ,
    ttl: 180
    # ( ) ,
    # loginInfo ( )
    activityTimeLimit: 180
  user:
    # ( )
    #
    allowedInactivePeriod: 180
  push:
    firebase:
      enabled: true
    apns:
      enabled: true

```

```

dictionary:
  notificationPath: "dictionary/notificationDict.yml"
  automationPath: "dictionary/automationDict.yml"
  guardPath: "dictionary/guardDict.yml"
  inputKeysPath: "dictionary/inputKeysDict.yml"
  deprecatedKeysPath: "dictionary/deprecatedKeysDict.yml"
  substitutionKeysPath: "dictionary/substitutionKeys.yml"

languageInterface:
  language: "ru"

# Hazelcast instance configuration -> Move it to separate microservice -> Use hz-client here
hazelcast:
  instanceName: "iot-core-hz-instance{{ containerNameSuffix }}"
  clusterName: "iot-core{{ containerNameSuffix }}"
  network:
    port: "5705"
#   member: "127.0.0.1:5701"

mjollnir:
  sync-period: "0 0 23 1/1 * ?"
  login: "platform"
  password: "platform"
  url: "{{ coreInternal.mjollnir.url }}"

#   iot-core
iot-core:
  host: "{{ mongodb_addr }}"
  port: {{ mongodb_port }}
  user: ""
  password: ""
  database: "{{ coreInternal.core.db.name }}"

#   iot-fs
file-storage:
  host: "{{ mongodb_addr }}"
  port: {{ mongodb_port }}
  user: ""
  password: ""
  database: "{{ coreInternal.fs.db.name }}"

#
license-storage:
  host: "{{ mongodb_addr }}"
  port: {{ mongodb_port }}
  user: ""
  password: ""
  database: "{{ coreInternal.licenses.db.name }}"

#   eventlog
eventlog:
  host: "{{ mongodb_addr }}"
  port: {{ mongodb_port }}
  user: ""
  password: ""
  database: "{{ coreInternal.events.db.name }}"

```

```
monitoring-system:
  countAttempt: 3
  delayAttempt: 0

spring:
  kafka:
    enabled: false
    producer:
      bootstrap-servers: "localhost:9092"
      key-serializer: org.apache.kafka.common.serialization.StringSerializer
      value-serializer: org.springframework.kafka.support.serializer.JsonSerializer

# ,      Android. ApiKey
video:
  flussonic:
    url: "{{ core.video.flussonic.url }}"
    apiKey: "{{ core.video.flussonic.apiKey }}"
    operator_id: "{{ core.video.flussonic.operatorId }}"
    admin_login: "{{ core.video.flussonic.adminLogin }}"
    fake_camera_url: "rtsp://wowzaec2demo.streamlock.net/vod/mp4:BigBuckBunny_115k.mp4"
  ivideon:
    # IP,  ivideon  .
    ip_for_catch_event: ""
  eltex_server:
    url: "{{ core.video.eltex_server.url }}"

#      SSL
ssl:
  key: "/etc/ssl/private/eltex-sc-api.key"
  crt: "/etc/ssl/certs/eltex-sc-api.crt"

proxy:
  enabled: false
  host: ""
  port: 8050

billing:
  cron: "0 0 0 * * ?"
  ftp:
    host: "127.0.0.1"
    port: 21
    login: "user"
    password: "password"
    workdir: "test"

notifications:
  duplicate_delay_sec: 0
  antispam_time_sec: 0
  delay_push_time_millisec: 250

guard:
  deviceRequestDelay: 15000
```

```

# swagger.      key      swagger
springdoc:
  swagger-ui:
    enabled: {{ 'true' if coreInternal.swagger.enable else 'false' }}
    key: ""

controller:
  delay_answer_from_control_millisec: 12000
  number_of_ping_threads: 1

oauth2:
  clientParameters:
    {{ core.yandexSkill.clientId }}:
      type: YANDEX
      password: "{{ core.yandexSkill.password }}"
      clientAuthenticationMethod: client_secret_post
      authorizationGrantTypes:
        - authorization_code
        - refresh_token
      redirectUri: "https://social.yandex.net/broker/redirect"
      additionalAccount: YandexAccount
    {{ core.sberSkill.clientId }}:
      type: SBER
      password: "{{ core.sberSkill.password }}"
      clientAuthenticationMethod: client_secret_post
      authorizationGrantTypes:
        - authorization_code
        - refresh_token
      redirectUri: "https://gateway.iot.sberdevices.ru/gateway/v1/binder/backward"
      additionalAccount: SberAccount
    {{ core.marusyaSkill.clientId }}:
      type: MARUSYA
      password: "{{ core.marusyaSkill.password }}"
      clientAuthenticationMethod: client_secret_post
      authorizationGrantTypes:
        - authorization_code
        - refresh_token
      redirectUri: "https://vc.go.mail.ru/smarthouse/{{ core.marusyaSkill.appId }}/callback"
      additionalAccount: MarusyaAccount

#      ()
# https://yandex.ru/dev/dialogs/smart-home/doc/concepts/about.html
yandex-skill:
  #      oauth2.clientParameters      client-id
  client-id: "{{ core.yandexSkill.clientId }}"
  #      API
  callback-uri: "https://dialogs.yandex.net/api/v1/skills"
  # ID ,      ( )
  skill-id: "{{ core.yandexSkill.skillId }}"
  # OAuth-,      ( )
  oauth-token: "{{ core.yandexSkill.oauthToken }}"

```

```

#
# https://developers.sber.ru/docs/ru/smarthome/overview
sber-skill:
#     oauth2.clientParameters    client-id
client-id: "{{ core.sberSkill.clientId }}"
#     API
callback-uri: "https://partners.iot.sberdevices.ru"
# Bearer-,      (      )
bearer-token: "{{ core.sberSkill.bearerToken }}"

#     Mail.ru
# https://help.mail.ru/marusia/smart_home
marusya-skill:
#     oauth2.clientParameters    client-id
client-id: "{{ core.marusyaSkill.clientId }}"
#     API
callback-uri: "https://vc.go.mail.ru/smarthouse/events/yandex"
# ID ,      (      )
app-id: "{{ core.marusyaSkill.appId }}"
# OAuth-,      (      )
oauth-token: "{{ core.marusyaSkill.oauthToken }}"

scriptengine:
#     executionNumberLimit
#     , scriptTimeLimit    scriptTimeLimit (),
#
frequentScript:
    executionNumberLimit: 60
    scriptTimeLimit: 60000
#     executionNumberLimit
#     , scriptTimeLimit    scriptTimeLimit (),
#
frequentScriptsInHouse:
    executionNumberLimit: 120
    scriptTimeLimit: 120000

logging:
    config: "classpath:logback-prod{{ ' ' if elk.enable else '-without-logstash' }}.xml"
    logback:
        dir: "/var/log/eltex-sc"
{% if elk.enable %}
    logstash:
        host: "{{ 'logstash' if iot.serverName == elk.serverName else elk.serverName }}"
        port: {{ 5000 if iot.serverName == elk.serverName else elkServices.logstash.port.map }}


```

```
{% endif %}
level:
  root: {{ core.logLevel }}
  org.springframework: WARN
  org.springframework.cache: WARN
  org.springframework.data: WARN
  org.springframework.web: WARN
  _org.springframework.web: WARN
  org.springframework.security: WARN
  org.springframework.security.oauth2: WARN
  org.springdoc: WARN
  org.mongodb: WARN
  org.eclipse.jetty: WARN
  org.apache.http: WARN
  org.hibernate: WARN
  org.jboss: WARN
  org.quartz.core.QuartzSchedulerThread: WARN
  io.swagger: WARN
  io.github.resilience4j: WARN
  io.netty: WARN
  io.mongock: WARN
  io.micrometer: WARN
  com.hazelcast: WARN
  com.hivemq: WARN
```

- , :
- **8069** — Ethernet TCP-;
 - **8070** — WS- ;
 - **8071** — HTTP- API-;
 - **8072** — WSS- ;
 - **8073** — HTTPS- API-;
 - **8883** — MQTT-;
 - **8088** — CAPTCHA.

Eltex SC . - , Eltex Home.
 : < **Eltex SC**>.

htop

 MEM% CPU% mongo eltex-sc.

df -h



Z-Wave Wi-Fi

```
netstat -na | grep 8070
netstat -na | grep 8883
```

```
, :
    • :
        • ESTABLISHED, LISTEN — ;
        • LAST_ACK — , IP;
        • TIME_WAIT, CLOSE_WAIT — , ;
    • 2- — ;
    • 3- — , ;
    • 5- — IP-.
```

- /storage/iot/core/var/log/eltex-sc/server*. (15 2022) , :

```
grep < > server-2022-01-15* > < >
```

```
, :
    • -i — ;
    • -n — ;
    • -h — ;
    • -A — ;
    • -B — .
```

```
:
grep -i -n -A 5 -B 2 error server-2022-01-15* > errors.log
```

```
:
    • ERROR;
    • ID IP;
    • PONG — , , .
```

-
- — (IP —).

API. URL - :

```
< Eltex SC>:< API>/api/v1/version
```

API — API HTTP (8071/8073). , API JSON.

```
{
  "version" : "1.26-3477",
  "api" : "1.0",
  "currentTime" : "2022-07-25T09:24:12.544842Z[Etc/UTC]"
}
```

http https



PKCS#8.

```
# head -1 /tmp/eltex-sc-api.key
-----BEGIN PRIVATE KEY-----

# head -1 /storage/iot/ssl/private/eltex-sc-api.key
-----BEGIN RSA PRIVATE KEY-----
```

- BEGIN PRIVATE KEY — PKCS#8;
- BEGIN RSA PRIVATE KEY — PKCS#1.

PKCS#1, PKCS#8.

PKCS#1 PKCS#8, letsencrypt:

```
openssl pkcs8 -topk8 -inform PEM -outform PEM -nocrypt -in ____ -out ____.
```

CAPTCHA

CAPTCHA . , - .

CAPTCHA `/etc/ansible-iot-1.26/templates/default-for-docker.yml.j2`

`/etc/ansible-iot-1.26/vars/service_params.yml`

`caseSensitive` .

`/etc/ansible-iot-1.26/vars/default.yml`

| | | |
|--------|---|---|
| | | |
| easy | . | . |
| medium | . | . |
| hard | . | . |