

# Настройка Panda для публикации приложений

**JumpServer** поддерживает использование как Windows Server, так и Linux в качестве машины для публикации приложений, например для публикации браузеров Chrome и Firefox для HTTP сессий и различных клиентов для работы с СУБД.

## Типы публикации приложений:

**Microsoft RemoteApp:** способ публикации приложений на базе Windows Server, обеспечивающий максимальную плавность работы. Требуется дополнительной настройке Windows Server и приобретения Microsoft RDS CALs.

**Panda (Виртуальное приложение):** способ публикации приложений на базе Linux, характеризующийся средней плавностью работы, хорошей совместимостью и поддержкой таких операционных систем, как CentOS, RedHat, Kylin и openEuler.

## Настройка Panda для публикации приложений.

### Принцип работы:

Машина для публикации приложений на базе операционной системы Linux использует контейнерную технологию, которая изолирует приложение в независимой среде выполнения. С помощью компонента Panda, предоставляемого JumpServer, осуществляется управление виртуальными приложениями.

### Процесс выглядит следующим образом:

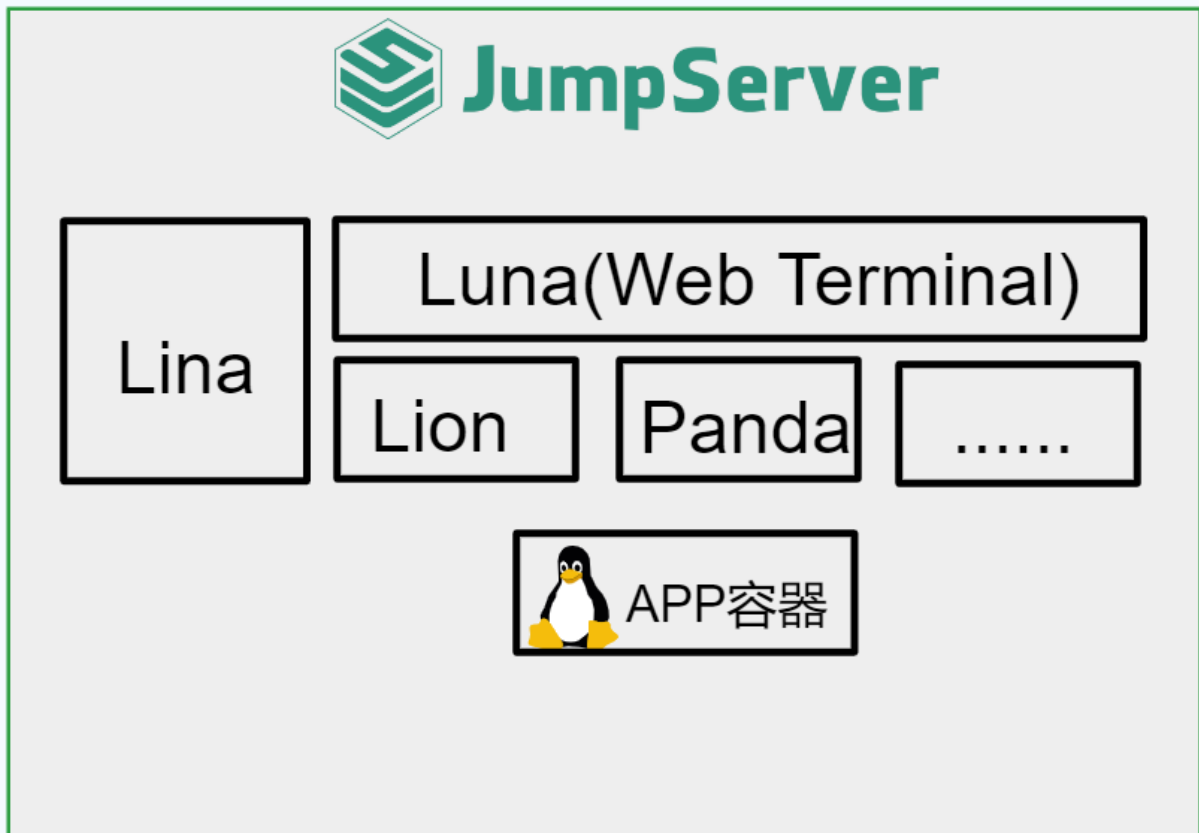
1. Пользователь получает доступ к Web Terminal JumpServer и подключается выбранному виртуальному приложению.
2. Компонент Panda создает GUI-контейнер на базе VNC и передает информацию о подключении VNC компоненту Lion.
3. Компонент Lion подключается к данному контейнеру.

## Схемы развертывания

### Схема 1: All in One

Использование сервера, на котором развернут JumpServer, в качестве машины для публикации виртуальных приложений.

192.168.127.162



## 1. Настройка основного конфигурационного файла

Откройте основной конфигурационный файл JumpServer.

```
nano /opt/jumpserver/config/config.txt
```

И добавьте в него следующие параметры

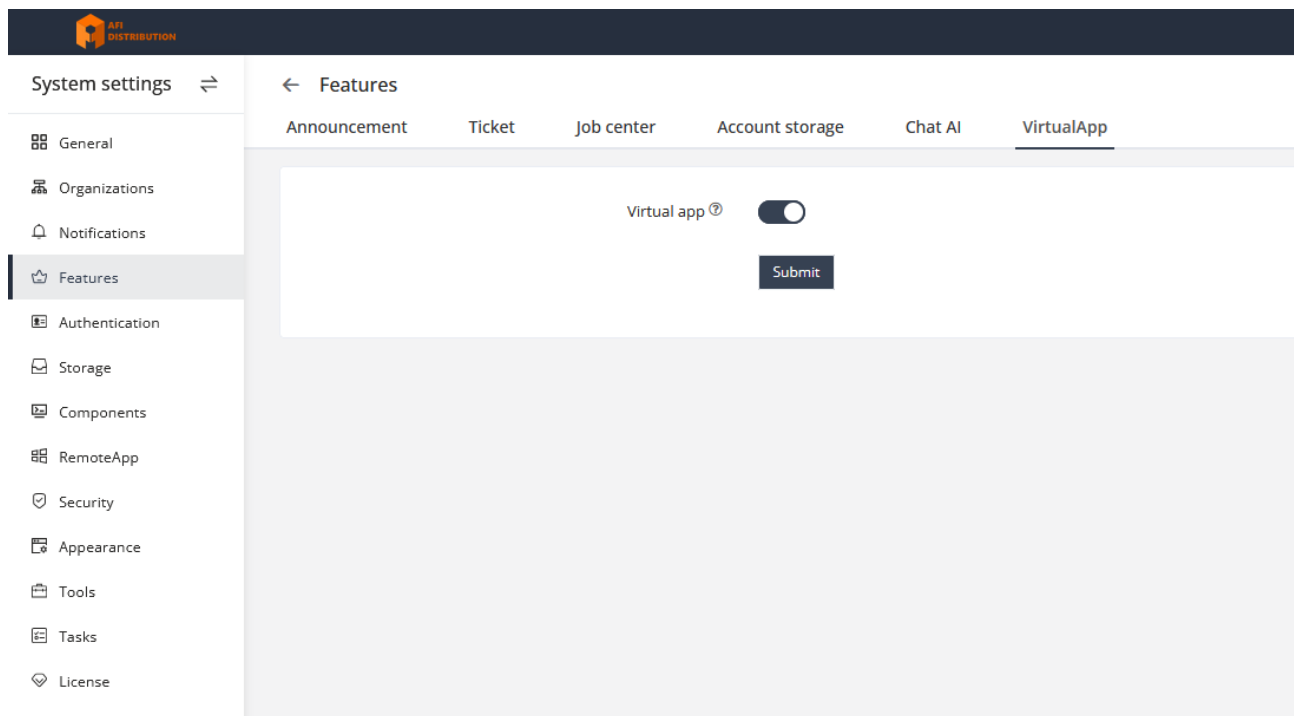
```
# Включение компонента Panda
PANDA_ENABLED=1
# Включение виртуальных приложений в ядре
VIRTUAL_APP_ENABLED=1
# IP-адрес хоста Panda (IP JumpServer)
PANDA_HOST_IP=192.168.127.162
# URL-адрес для компонента Lion к Panda
PANDA_HOST=http://panda:9001
```

Перезапустите сервис JumpServer, чтобы применить изменения.

```
[root@localhost ~]# jmsctl restart
```

## 2. Включение функции виртуальных приложений

В консоли управления JumpServer перейдите в **System Settings** → **Features** → **VirtualApp** и активируйте функцию виртуальных приложений.

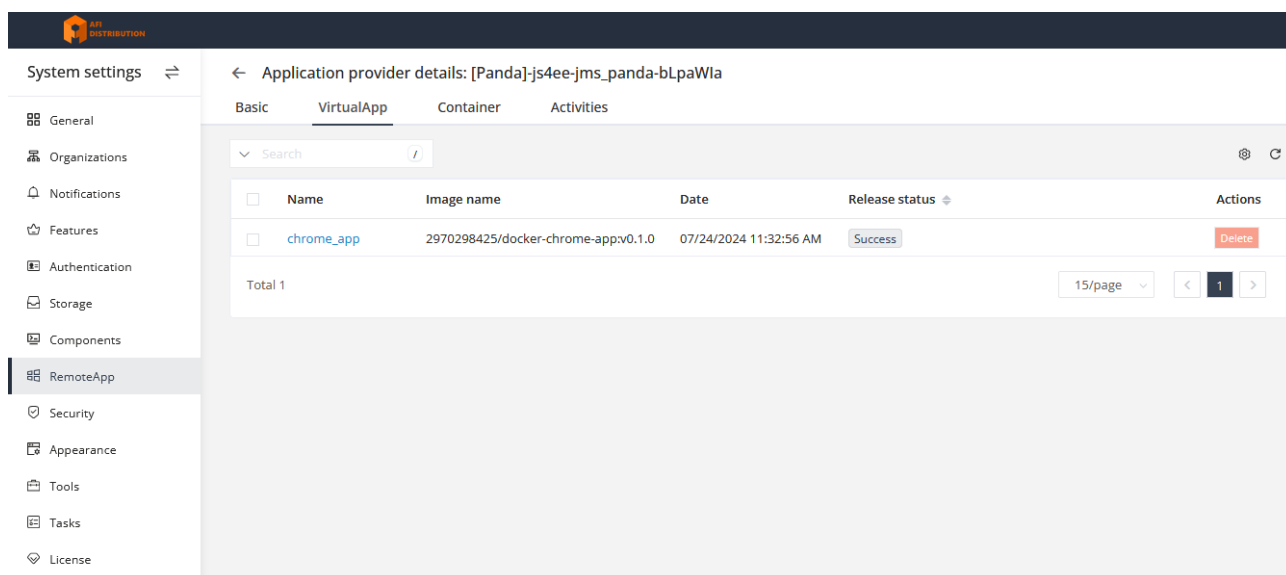


### 3. Загрузка виртуальных приложений

Загрузите виртуальные приложения локально. В настоящее время поддерживаются браузер **Chrome**, клиент базы данных **Dbeaver**. **Дистрибутивы этих приложений доступны на портале вендора, приложения для Panda находятся в разделе Virtual App, остальные апплеты только для RemoteApp(RDS).**

В консоли управления **JumpServer** перейдите в **System Settings** → **RemoteApps** и загрузите виртуальные приложения в раздел **VirtualApp**.

После короткого ожидания приложение будет автоматически развернуто на машине для публикации приложений. В консоли управления **JumpServer** в разделе **System Settings** → **RemoteApps** → **Application Providers** → **VirtualApp** можно увидеть успешное развертывание приложения.

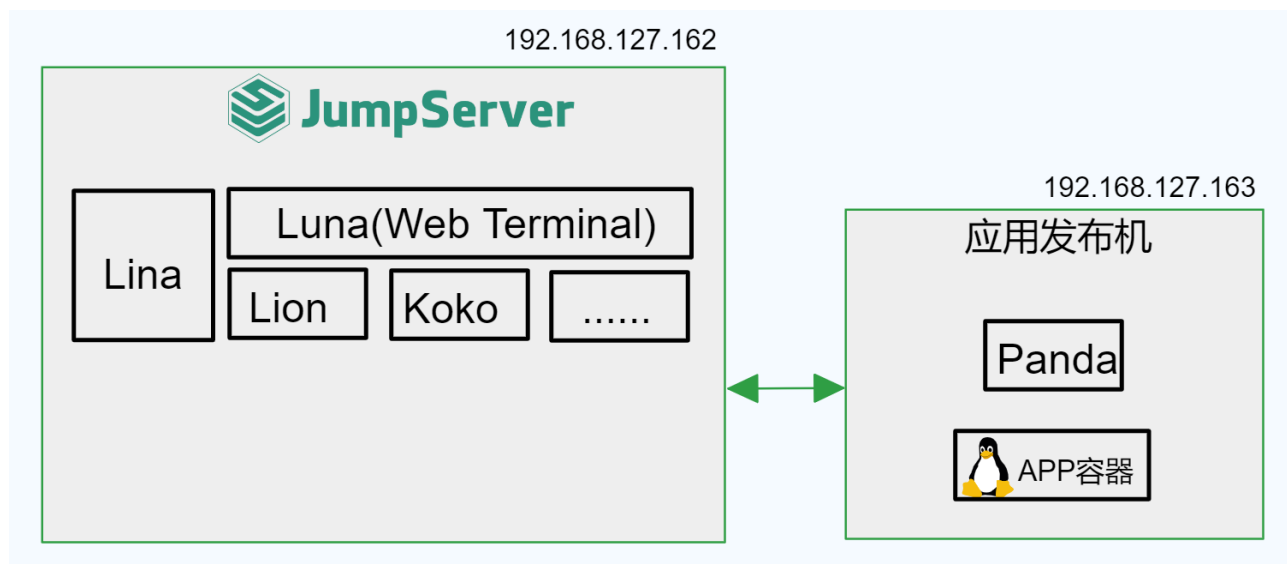


#### 4. Использование виртуальных приложений

Подключитесь к активам, используя виртуальные приложения.

**Примечание:** В этот момент в сервисе JumpServer будет запущен контейнер виртуального приложения: **2970298425/docker-chrome-app:v0.1.0** (Примечание: этот контейнер весит около 1.3GB, требуется загрузка через интернет. В локальной сети можно загрузить его вручную).

### Схема 2: Panda на другом сервере



#### 1. Настройка основного конфигурационного файла

Откройте основной конфигурационный файл JumpServer.

```
nano /opt/jumpserver/config/config.txt
```

И добавьте в него следующие параметры

```
# Включение компонента Panda
PANDA_ENABLED=0
# IP-адрес Panda для компонента Lion
PANDA_HOST=http://192.168.127.163:9001
```

Перезапустите сервис JumpServer, чтобы применить изменения.

```
[root@localhost ~]# jmsctl restart
```

## 2. Установка Panda на сторонней машине

Распакуйте установочный пакет JumpServer на машине для публикации, установите Docker и Docker Compose, загрузите образ.

```
[root@panda ~]# tar xzvf jumpserver-offline-release-v3.10.6-amd64.tar.gz -C /opt
```

Установите Docker и Docker Compose:

```
[root@panda ~]# cd /opt/jumpserver-offline-release-v3.10.6-amd64/scripts
[root@panda scripts]# ./2_install_docker.sh
```

Загрузите образ Panda:

```
[root@panda scripts]# cd images
[root@panda images]# docker load -i panda:v3.10.6.tar
```

Создайте docker-compose для Panda:

```
[root@panda ~]# mkdir -p /data/jumpserver/panda/data
[root@panda ~]# mkdir -p panda
[root@panda ~]# cd panda
[root@panda panda]# cat docker-compose.yaml
version: '2.4'

services:
  panda:
    image: registry.fit2cloud.com/jumpserver/panda:v3.10.6
    container_name: jms_panda
    hostname: jms_panda
    ulimits:
      core: 0
    restart: always
    ports:
      - 9001:9001
    tty: true
    environment:
      - BOOTSTRAP_TOKEN=YmEyNTRkNTYtNDIyMi02OTJm
      - CORE_HOST=http://192.168.127.162
      - NAME=panda
```

```
- PANDA_HOST_IP=192.168.127.163
volumes:
- /data/jumpserver/panda/data:/opt/panda/data
- /var/run/docker.sock:/var/run/docker.sock:z
healthcheck:
test: "curl -fsL http://localhost:9001/panda/health/ > /dev/null"
interval: 10s
timeout: 5s
retries: 3
start_period: 10s
```

**BOOTSTRAP\_TOKEN** берется из файла конфигурации JumpServer: /opt/jumpserver/config/config.txt

**CORE\_HOST** - адрес вашего JumpServer

**PANDA\_HOST\_IP** - IP адрес Panda

Запустите контейнер Panda:

```
docker-compose up -d
```

### 3. Включение функции виртуальных приложений

Повторите шаги из раздела All in One

### 4. Загрузка виртуальных приложений

Повторите шаги из раздела All in One

### 5. Использование виртуальных приложений

Повторите шаги из раздела All in One

---

Версия #7

Сергей Попцов создал 30 августа 2024 16:26:40

Сергей Попцов обновил 9 декабря 2024 16:23:28