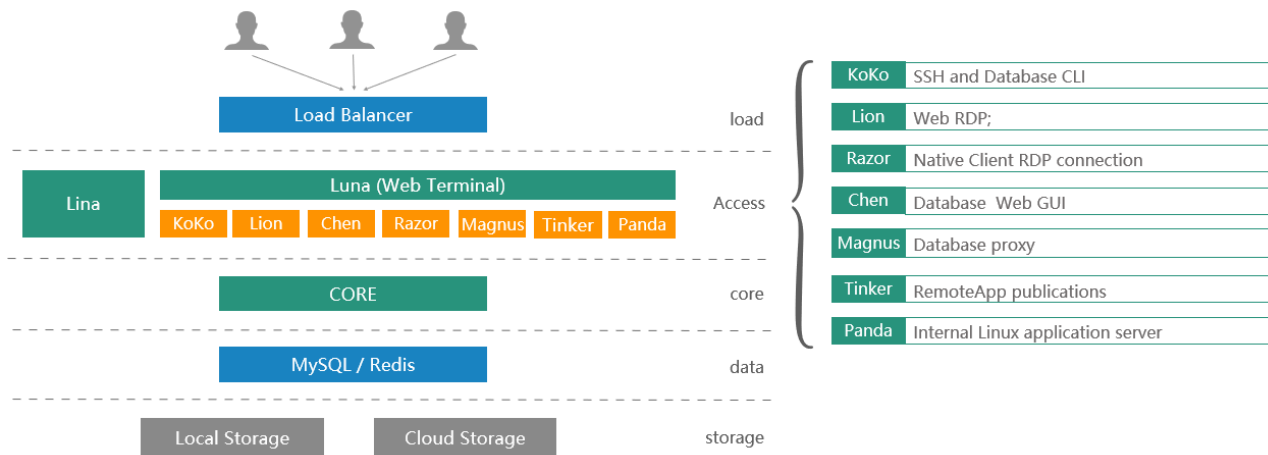


Troubleshooting

- [How to check system status and container logs](#)
- [RemoteApp Troubleshooting](#)

How to check system status and container logs

JumpServer is installed as a set of Docker containers performing various functions.



To check the state of the containers, log in to the server where JumpServer is installed and enter the following command:

```
# docker ps -a
```

All containers in the list should have the status **Healthy**

List of Containers

```
jms_panda
jms_magnus
jms_celery
jms_chen
jms_koko
jms_lion
jms_razor
jms_video
jms_web
jms_redis
jms_kael
jms_xrdp
jms_mysql
jms_core
```

To view the error log of a specific module, use the following command:

```
docker logs -f #CONTAINER_NAME --tail 200
```

For example:

Web interface error log:

```
docker logs -f jms_web --tail 200
```

Error log for Panda:

```
docker logs -f jms_panda --tail 200
```

RemoteApp Troubleshooting

Description of the Interaction Between JumpServer, RDS Server, and Tinker

In this article, I will describe in detail how JumpServer, RDS Server, and Tinker interact so that any issues can be identified.

Description of the RDS and JumpServer Integration Process

1. JumpServer connects to the RDS server via SSH:

- Installs the Tinker service on the RDS server and specifies the JS address (Core API parameter).
- Creates local service accounts of the type `js_*` and `jms_*` on the RDS server.
- Adds the created accounts to the "Remote Desktop Users" group.

“ Important: The "Remote Desktop Users" group name may differ in Windows, depending on the OS language. If these accounts are not present in the group, you will need to add them manually.

1. Tinker connects to JumpServer at the specified address (Core API parameter):

- Tinker reports that it is running, and the RDS status changes to "online".
- Downloads distributions of selected applets, such as Chrome and DBeaver, from the JS server using the same address specified in the Core API. Other applets may be downloaded from the internet. If there is no internet access, you can manually install the required applications.
- Tinker installs the downloaded applets

Description of Tinker

Tinker is an auxiliary application for integration with JumpServer, assisting in maintaining communication between RDS and JS, installing applications (applets) required for publication, and managing the publication of RemoteApp applications during session initiation.

- By default, it is installed in the folder:
`C:\Users\[ACCOUNT]\AppData\Local\Programs\Tinker\`
(`[ACCOUNT]` is the account used for the integration of JS and RDS.)
- It installs the **JumpServer Tinker Service** - this service must always be running.

- Detailed logs of Tinker's operation and errors are available in the folder:

C:\Users\[ACCOUNT]\AppData\Local\Programs\Tinker\data\logs

Tinker Applets

Applets are a set of applications and automation scripts for managing applications. By default, applets are installed in the folder:

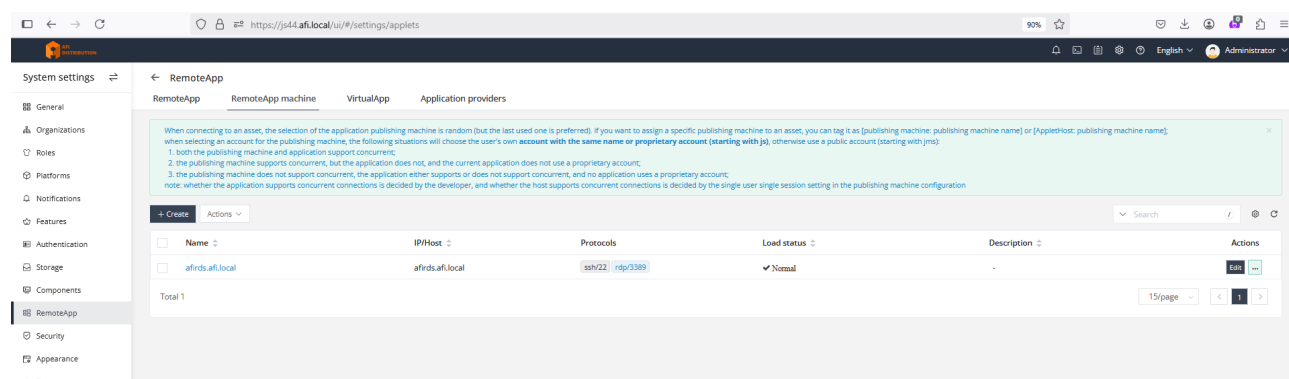
C:\Program Files\JumpServer

This folder contains executable applications and Python scripts with launch parameters that you can modify if necessary.

Final Checklist

- **OpenSSH** is installed on Windows RDS, and port **22** is open on the Windows firewall.
- The account used for integration with RDS has **administrator rights** on the RDS server.
- The **IP/Host** field contains the **IP address** of the RDS server or its **DNS name**, which can be resolved by JumpServer.
- The **Core API** field contains the **URL of JumpServer**, accessible from the RDS Server.
- Accounts of the type `JS_XX` and `JMS_XX` are created on the RDS server.
- Accounts of the type `JS_XX` and `JMS_XX` are members of the **Remote Desktop Users** group on the RDS server.
- The **JumpServer Tinker Service** is running on the RDS server.
- The **RDS server status** in the JumpServer interface: **Normal**.
- The **Status of applets** in the properties of the RDS server in the JumpServer interface: **Success**.

Here are some screenshots for example:



Basic

* Name

* IP/Host

Protocol

Protocols

ssh	22		
rdp	3389		

The protocols supported by the assets are restricted by the platform. Click the settings button to view the protocol settings. If updates are required, please update the platform

Account

Accounts [Update account info in asset details](#)

Using same account ☐

Auto create accounts ☒

Accounts create amount

Automation

Core API

Ignore certificate verification ☒

Existing RDS license ☐

System settings

General

Organizations

Roles

Platforms

Notifications

Features

Authentication

Storage

Components

RemoteApp

Security

Appearance

Tools

System tasks

License

← RemoteApp

RemoteApp

RemoteApp machine

VirtualApp

Application providers

When connecting to an asset, the selection of the application publishing machine is random (but the last used one is preferred). If you want to specify when selecting an account for the publishing machine, the following situations will choose the user's own account with the same name or password:

1. both the publishing machine and application support concurrent;

2. the publishing machine supports concurrent, but the application does not, and the current application does not use a proprietary account;

3. the publishing machine does not support concurrent, the application either supports or does not support concurrent, and no application uses a proprietary account.

note: whether the application supports concurrent connections is decided by the developer, and whether the host supports concurrent connections is decided by the platform.

Create

Actions

<input type="checkbox"/>	Name	IP/Host	Protocols
<input type="checkbox"/>	afirds.afi.local	afirds.afi.local	ssh:22 rdp:3389

Total 1

RemoteApp machine: afirds.afi.local

Basic

Accounts

RemoteApp

Deploy publishing machine

Activities

RECENT (7 DAYS)

Recently discovered | Recently logged in | Recently modified | Recently changed password | Recent password change failed

RISKY ACCOUNT

No login for long time | Add account | Weak password | Empty password | Long time password

ACCOUNT TYPE

All | Host | Database | Cloud | Device | Web | Other

Create

Account templates

Actions

Search

1

Filter

Download

Refresh

<input type="checkbox"/>	Name	Username	Asset	Connect	Actions
<input type="checkbox"/>	jms_5VODdjer	jms_5VODdjer	afirds.afi.local		
<input type="checkbox"/>	jms_62IQmV1	jms_62IQmV1	afirds.afi.local		
<input type="checkbox"/>	jms_D66i2Sp5	jms_D66i2Sp5	afirds.afi.local		
<input type="checkbox"/>	jms_EKj8v3N	jms_EKj8v3N	afirds.afi.local		
<input type="checkbox"/>	jms_Ear7CsCq	jms_Ear7CsCq	afirds.afi.local		
<input type="checkbox"/>	jms_JRXD9cYj	jms_JRXD9cYj	afirds.afi.local		
<input type="checkbox"/>	jms_O58h7ro4	jms_O58h7ro4	afirds.afi.local		
<input type="checkbox"/>	jms_YWj71Nn2	jms_YWj71Nn2	afirds.afi.local		
<input type="checkbox"/>	jms_ppOf6i3m	jms_ppOf6i3m	afirds.afi.local		
<input type="checkbox"/>	jms_wQ000Ri	jms_wQ000Ri	afirds.afi.local		
<input type="checkbox"/>	js_ACUNETIX	js_ACUNETIX	afirds.afi.local		
<input type="checkbox"/>	js_ACUNETIX8	js_ACUNETIX8	afirds.afi.local		
<input type="checkbox"/>	js_AFIDC	js_AFIDC	afirds.afi.local		
<input type="checkbox"/>	js_AFIDC8	js_AFIDC8	afirds.afi.local		

Basic Accounts RemoteApp Deploy publishing machine Activities

Search



<input type="checkbox"/>	Name	Version	Date	Release status	Actions	
<input type="checkbox"/>	Base	0.2.2	04/02/2025 11:24:19	Success	Deploy...	Uninst...
<input type="checkbox"/>	Mong	0.2.2	04/02/2025 11:24:36	Success	Deploy...	Uninst...
<input type="checkbox"/>	Firefc	0.2.0	04/02/2025 11:25:04	Success	Deploy...	Uninst...
<input type="checkbox"/>	DBea	0.3	04/02/2025 11:25:40	Success	Deploy...	Uninst...
<input type="checkbox"/>	Chro	1.1	04/02/2025 11:25:40	Success	Deploy...	Uninst...

Total 5

15/page

< 1 >

Quick update

Initialization deployment

Deployment

Only initial deployment

Deployment

Publish all applications

Publish