PlayceRoRo
통합 설치가이드

오픈소스컨설팅
OPEN SOURCE CONSULTING

Table of Contents

1장 설치 전 준비 5

Playce RoRo 통합 패키지 소개 6

설치 완료 예상 시간 6

설치 전 준비 사항 6

필수 설치 소프트웨어 7

Playce RoRo Server 7

Inventory Server(Rehost migration대상) 8

Inventory Server(Replatform migration 대상) 8

웹 클라이언트(브라우저) 9

시스템 사양 10

권장 사양 10

Playce RoRo 사용자/작업 디렉터리 생성 11

RoRo Server 11

Inventory Server 11

2장 설치하기 12

설치 파일 다운로드 13

설치 파일 압축 해제 14

압축 해제 패키지 설치 14

루트(root) 계정으로tar.gz 파일 압축 해제 14

로로(roro) 계정으로tar.gz 파일 압축 해제 15

디렉터리 구조 16

playce 디렉터리 구조 16

playce-roro, playce-migrator 디렉터리 구조 17

환경 설정 18

기본 포트에 설치하는 경우 18

기본 게이트웨이 포트 변경 예시 19

Playce 제품군의 하위 디렉터리 포트 변경 19

Playce RoRo for Java App의 작업 디렉터리 변경 20

데이터베이스 환경 구성 21

타임존 설정 21
Database 및 User 생성 22
Playce RoRo 패치 23
패치 파일 다운로드 23
3.5.1 버전에서 3.6.0으로 패치하기 23
패치 작업 Flow 24
Playce Gateway 변경 사항 25

3장 실행하기 27
서비스 실행 28
Java 버전 확인 28
Java 설치 28
설치 확인 29
방화벽 설정 수정 30

4장 시작하기 31
로그인 32
비밀번호 초기화 34
비밀번호 변경 35
비밀번호 작성 규칙 37

Playce RoRo | 통합 설치가이드

Copyright © 오픈소스컨설팅, Inc. All rights reserved.

Playce RoRo 통합 설치가이드의 내용과 프로그램은 저작권법에 의해서 보호받고 있습니다. 이 문서에 표기된 제품명은 각 사의 등록상표입니다.

제품 정보

- 소프트웨어 버전: Playce RoRo v3.7.0 & Playce RoRo for Java App v1.0.0
- 문서 버전: Version 1.2.0
- 발행 날짜: 2023년 8월 11일

기술지원 정보

- 담당부서: 오픈소스컨설팅 Playce Dev팀
- 이메일: lab@osci.kr
- 기술지원: 070-4268-0995
- 전화: 02-516-0711
- 팩스: 02-516-0722
- 홈페이지: http://www.osci.kr
- 주소: 서울특별시 강남구 테헤란로 83길 32, 5층 (삼성동, 나라키움삼성동A빌딩)

1장 설치 전 준비

Playce RoRo 통합 패키지 소개

Playce RoRo 통합 패키지는 인벤토리, 서버, 미들웨어, 자바 애플리케이션, 데이터베이스를 스캔하고 분석하는 기능을 가진 Playce RoRo와 분석 규칙을 기반으로 자바 애플리케이션의 소스 코드를 분석하고 마이그레이션 대상 환경에 이전이 가능하도록 소스 코드를 변경하는 Playce RoRo for Java App 2개 제품을 한 번에 설치할 수 있는 버전입니다.

Playce RoRo 통합 패키지는 설치 편의성 제공과 더불어 안전한 로그인과 상호 인증을 위한 인증 모듈과 각 연결을 제어하는 프록시 모듈로 구성되어 있습니다.

설치 완료 예상 시간

Playce RoRo 통합 설치가이드에 따라 작업을 진행할 경우 예상 소요 시간은 30분 정도로 예상됩니다.

설치 전 준비 사항

Playce RoRo 통합 패키지는 수집/분석된 데이터를 저장하고 관리하는 데이터베이스가 필요합니다. 또한 자반 기반의 프로그램 작동 및 활용을 위해 Java 11.0.13 이상 버전이 사전에 설치되어 있어야 합니다.

Playce RoRo 통합 패키지 설치를 위한 하드웨어 및 소프트웨어 요구 사항은 다음과 같습니다.

- CPU, 메모리 및 하드디스크 공간이 확보된 Cent OS 7 이상 또는 Rocky Linux 9 이상
- JDK(Java Development Kit) 11.0.13 버전 이상
- MariaDB 10.6 이상

필수 설치 소프트웨어

Playce RoRo 설치 전 반드시 설치 대상 서버에 먼저 설치되어 있어야 하는 필수 소프트웨어입니다. 설치 대상 서버가 외부 인터넷 접속이 가능한 경우에는 wget을 사용하여 직접 다운로드할 수 있습니다.

- tar 패키지

Playce RoRo Server

- Java JDK 11.0.13

참고

JRE는 Application Assessment 수행 시 일부 항목이 수집되지 않을 수 있습니다.

- Python 2.7
- System Package
  - ssh
  - rsync
  - epel
  - sshpass
- Maria DB 10.6 이상

참고

인벤토리 및 분석 데이터 저장을 위한 데이터베이스로 사용되며 커뮤니티 버전이 설치 가능한 MariaDB를 설치해야 합니다.

- /etc/mke2fs.conf 파일의 [fs_types] 영역 ext4의 features 인자를 다음과 같이 설정하면 Rehost Migration 실패에 대비 가능

```
[defaults]
    ... skip ...

[fs_types]
    ... skip ...
    ext4 = {
        features = has_journal,extent,huge_file,flex_bg,metadata_csum,dir_nlink,extra_isize
```

```
        auto_64-bit_support = 1
        inode_size = 256
    }
    ... skip ...
```

Inventory Server(Rehost migration 대상)

- System Packages
  - ssh
  - rsync
- 등록할 사용자 계정이 root가 아닌 경우 /etc/sudoers 파일에 추가할 내용

```
${USER_NAME} ALL=(ALL) NOPASSWD: ALL
```

위와 같이 설정하면, 해당 사용자 또는 그룹이 패스워드 없이 sudo 명령 실행이 가능해집니다.

- /etc/sudoers 파일에 Defaults requiretty 옵션이 활성화되어 있는 경우 주석 처리

참고

OS 버전에 따라 해당 옵션이 존재하지 않을 수 있습니다.

Inventory Server(Replatform migration 대상)

- System Packages
  - ssh
  - sudo(Unix만 해당됨)
- 등록할 사용자 계정이 root가 아닌 경우 /etc/sudoers 파일에 추가할 내용

```
${USER_NAME} ALL=(ALL) NOPASSWD: ALL
```

위와 같이 설정하면, 해당 사용자 또는 그룹이 패스워드 없이 sudo 명령 실행이 가능해집니다.

- /etc/sudoers 파일에 Defaults requiretty 옵션이 활성화되어 있는 경우 주석 처리

참고

OS 버전에 따라 해당 옵션이 존재하지 않을 수 있습니다.

웹 클라이언트(브라우저)

- Chrome 47.x 이상의 최신 버전
- Microsoft Edge108.x 이상
- Microsoft Excel 2013 이상(Export/Import Data File용)

⚠ 주의

Internet Explorer 11 버전 사용은 권장하지 않습니다. 해당 브라우저에서는 Playce RoRo 통합 패키지 설치 테스트가 진행되지 않아 정상적인 작동을 보장할 수 없습니다.

시스템 사양

Playce RoRo 통합 패키지를 설치하기 전 시스템 권장 사양을 먼저 확인한 후, 사양에 맞는 시스템을 미리 준비하시기 바랍니다.

권장 사양

Playce RoRo 통합 패키지 설치를 위한 시스템 권장 사양은 다음과 같습니다.

- CPU: 4 Core
- 메모리: 8 GB
- 하드디스크: 50 GB 이상
- 운영 체제: Cent OS 7 이상
- JDK 버전: 11.0.13 이상

참고

하드디스크 용량은 Assessment 데이터 및 Migration 이미지 저장을 위해 필요합니다.

참고

JDK 11.0.13 미만 버전에서는 Security 및 ModelMapper 관련 오류가 있으므로 반드시 11.0.13 이상 버전을 설치하시기 바랍니다.

참고

Playce RoRo의 정상 작동을 위해 Java 플랫폼은 JRE(Java Runtime Environment)가 아닌 반드시 JDK(Java Development Kit)를 설치해야 합니다

Playce RoRo 사용자/작업 디렉터리 생성

Playce RoRo 통합 패키지를 설치하기 전 설치 대상 서버에 Playce RoRo 사용자 계정 및 작업 디렉터리를 만들어야 합니다. 새로 생성한 사용자 계정으로 Playce RoRo를 실행해 주시기 바랍니다.

본 문서에서는 사용자 계정으로 roro, 작업 디렉터리 이름도 roro로 생성합니다.

RoRo Server

Playce RoRo를 설치할 서버에서 사용자 계정과 작업 디렉터리를 생성합니다.

- 사용자 계정(roro) 생성

```
[root ~]# useradd roro
```

- 작업 디렉터리(/roro) 생성

```
[root ~]# mkdir -p /roro

[root ~]# chown roro.roro /roro
```

Inventory Server

Inventory Server로 사용할 서버에서는 sudo 명령 실행이 가능하도록 다음과 같이 설정합니다.

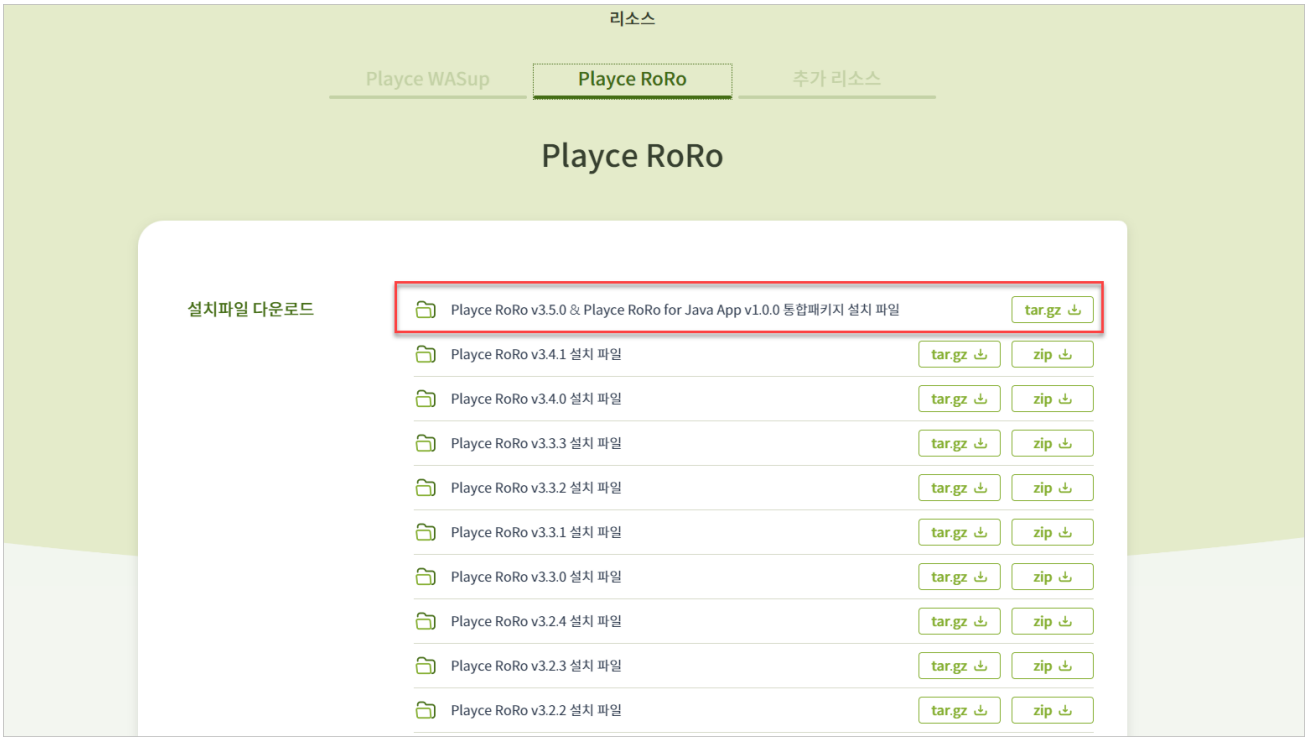
```
# Disable "ssh hostname sudo <cmd>", because it will show the password in clear.

#             You have to run "ssh -t hostname sudo <cmd>".

#

#Defaults      requiretty

.....

## Same thing without a password

# %wheel            ALL=(ALL)           NOPASSWD: ALL

roro      ALL=(ALL)           NOPASSWD: ALL
```

2장 설치하기

설치 파일 다운로드

Playce RoRo 통합 설치 파일은 다음의 경로에서 최신 버전을 다운로드할 수 있습니다.

- https://www.play-ce.io/resource.html#playce-roro



참고

Playce RoRo 통합 설치 파일은 tar 파일만 다운로드할 수 있습니다.

설치 파일 압축 해제

설치 파일을 다운로드한 후 Playce RoRo 서버로 사용할 대상 서버에 설치 파일을 업로드하고 파일의 압축을 해제해야 합니다.

압축 해제 패키지 설치

다운로드한 파일은 tar로 압축을 해제해야 합니다. 압축 해제 패키지가 설치되어 있지 않은 경우, 다음의 명령어를 실행하십시오.

```
# tar 패키지 설치
 ]$ sudo yum install -y tar
```

루트(root) 계정으로 tar.gz 파일 압축 해제

압축 해제 권장 디렉터리는 /opt입니다. /opt 디렉터리에 설치하기 위해서는 초기 root 권한으로 압축을 해제하고 권한을 변경해야 합니다.

> **참고**
>
> 다른 디렉터리에 환경을 구성하는 경우 사용자 계정 roro를 생성한 후에 생성한 계정으로 실행하는 것을권장합니다.

1. https://www.play-ce.io/resource.html에서 파일을 다운로드합니다.
2. 다운로드한 파일을 설치 대상 서버에 업로드합니다.
3. 압축 해제를 위해 다음의 명령어를 실행합니다.

```
# playce-roro-ext-{VERSION}.tar.gz 파일을 /opt에 설치하는 경우(playce라는 디렉터리가 자동으로 생성됨)
[root ~]# tar xf playce-roro-ext-{VERSION}.tar.gz -C /opt

[root ~]# chown -R roro:roro /opt/playce
```

- 하위 playce 디렉터리가 ${PLAYCE_HOME} 경로가 됨
- ${PLAYCE_HOME} 경로에는 bin, modules, playce-roro, playce-migrator 디렉터리가 있음
- ${PLAYCE_HOME}/playce-roro 디렉터리가 ${RORO_HOME}이 됨
- ${PLAYCE_HOME}/playce-migrator가 ${MIGRATOR_HOME} 경로가 됨

로로(roro) 계정으로 tar.gz 파일 압축 해제

root 계정이 아닌 roro 계정으로 설치를 진행하는 경우, sudoer에 등록되어 있다면 다음의 명령으로도 압축을 해제할 수 있습니다.

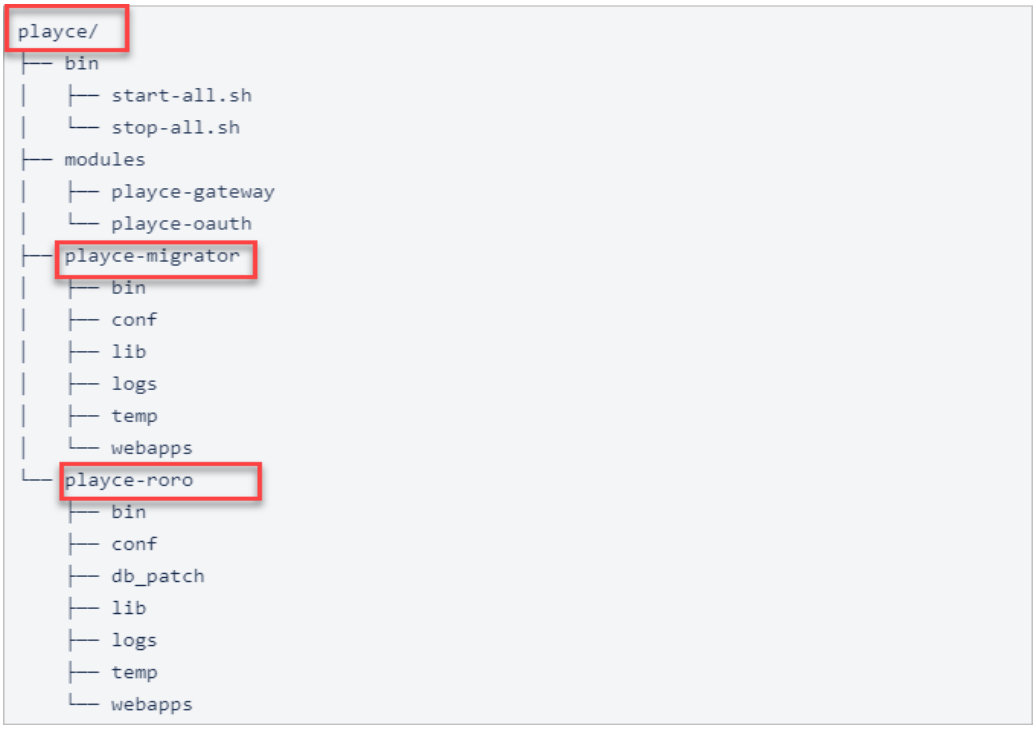
```
[roro ~]$ sudo mkdir -p /opt/playce

[roro ~]$ sudo chown roro.roro /opt/playce

[roro ~]$ tar xf playce-roro-ext-{VERSION}.tar.gz --strip-components=1 -C /opt/playce
```

디렉터리 구조

Playce RoRo 통합 패키지의 압축을 해제하면 다음과 같은 디렉터리 구조를 확인할 수 있습니다.

playce 디렉터리 구조

playce의 디렉터리 구조는 다음과 같으며, 각각의 디렉터리는 개별 서버입니다. playce 제품은 개별 서버 간 통신을 통해 구동됩니다. 각각의 디렉터리내에 bin/setenv.sh에서 서버별 데이터 베이스 설정, 작업 중 사용될 디렉터리, 로그 파일 위치를 설정합니다.



- playce/bin: 시작, 종료 및 기타 기능을 하는 스크립트 파일이 위치하는 디렉터리로 *.sh 파일 사용
- playce/modules/playce-oauth: 통합 로그인 관련 디렉터리와 파일이 위치하고 있는 디렉터리
- playce/modules/playce-gateway: playce 제품의 API 게이트웨이 관련 디렉터리와 파일이 위치하고 있는 디렉터리
- playce/playce-migrator: Playce RoRo for Java App 관련 디렉터리와 파일이 위치하고 있는 디렉터리
- playce/playce-roro: Playce RoRo 관련 디렉터리와 파일이 위치하고 있는 디렉터리

playce-roro, playce-migrator 디렉터리 구조

playce 디렉터리의 각 하위 디렉터리는 모두 동일한 구조의 하위 디렉터리로 구성되어 있습니다.

```
playce-migrator
    |- bin
    |- conf
    |- lib
    |- logs
    |- temp
    |- webapps
    |- work


playce-roro
    |- bin
    |- conf
    |- lib
    |- logs
    |- temp
    |- webapps
    |- work
```

- /bin: 시작, 종료 및 기타 기능을 하는 스크립트 파일이 위치하는 디렉터리로 *.sh 파일 사용
- /conf: 컨테이너 기본 구성 파일로 가장 중요한 server.xml 파일과 설정 파일이 위치하는 디렉터리
- /lib: 톰캣 등의 제품 라이브러리 관련 파일이 있는 디렉터리
- /logs: 기본 로그 파일이 저장된 디렉터리
- /temp: 기능 작동 시 필요한 파일이 위치하는 디렉터리
- /webapps: Playce RoRo 또는 Playce RoRo for Java App 웹 애플리케이션이 위치하는 디렉터리
- /work: 제품 실행 시 생성되는 파일을 임시 저장하거나 웹 애플리케이션 배포용 디렉터리

환경 설정

Playce RoRo를 설치하려면 다음의 내용에 따라 설치 대상 시스템에서 환경 설정을 해야 합니다.

기본 포트에 설치하는 경우

Playce RoRo의 기본 설치 포트와 환경 설정 파일은 다음과 같습니다.

- Gateway 서버: 8080 포트
- auth 서버: 8081 포트
- Playce RoRo: 8082포트
- Playce RoRo for Java App: 8083 포트
- 환경 설정 파일
  - playce/playce-migrator/bin/setenv.sh
  - playce/playce-roro/bin/setenv.sh

기본 OFFSET 설정

Playce RoRo(playce-roro), Playce RoRo for Java App(playce-migrator)의 환경 설정 파일인 setenv.sh 파일에 설정된 기본 OFFSET은 다음과 같습니다.

- Playce RoRo의 OFFSET

```
# Port offset for HTTP

PORT_OFFSET=2
```

  - 기본 Gateway 서버의 8080 포트에 OFFSET 값 2를 더한 8082 포트를 사용함

- Playce RoRo for Java App의 OFFSET

```
# Port offset for HTTP

PORT_OFFSET=3
```

  - 기본 Gateway 서버의 8080 포트에 OFFSET 값 3을 더한 8083 포트를 사용함

기본 게이트웨이 포트 변경 예시

Playce 제품군의 API 게이트웨이 기본 포트는 8080입니다. 해당 포트를 9080번 포트로 변경하려면 다음과 같이 설정합니다.

- 환경 설정 파일: playce/module/playce-gateway/bin/setenv.sh

```
# Port offset for HTTP

PORT_OFFSET=1000
```

Playce 제품군의 하위 디렉터리 포트 변경

기본 게이트웨이를 위의 예시와 같이 9080으로 변경하더라도 Playce 제품군의 하위 디렉터리에 있는 playce-outh, playce-migrator, playce-roro의 포트는 설정된 OFFSET인 8080을 기준으로 작동되므로 같은 형태로 변경하기 위해서는 다음과 같이 OFFSET을 변경해야 합니다.

- playce-outh: 9081번 포트로 변경

```
# Port offset for HTTP

PORT_OFFSET=1001
```

- playce-roro: 9082번 포트로 변경

```
# Port offset for HTTP

PORT_OFFSET=1002
```

- playce-migrator: 9083번 포트로 변경

```
# Port offset for HTTP

PORT_OFFSET=1003
```

참고

포트 충돌로 인한 경우를 제외하고는 Playce RoRo 통합 패키지의 기본 포트를 변경하는 것을 권장하지 않습니다.

참고

playce-oauth, playce-roro, playce-migrator의 setenv.sh 내의 포트를 변경하면 자동으로 gateway에서 해당 포트로 포워딩하므로 별도의 추가 설정이 필요 없습니다.

Playce RoRo for Java App 의 작업 디렉터리 변경

Playce RoRo for Java App(playce-migrator)의 기본 작업 디렉터리는 /opt/migrator로 설정되어 있습니다.

현재 통합 설치 가이드 문서의 설치 디렉터리가 /opt/playce로 변경되는 경우 Playce RoRo for Java App의 디렉터리를 실제 Playce RoRo for Java App이 설치된 디렉터리로 변경해야 합니다.

- 변경 대상 파일 위치: ${PLAYCE_HOME}/playce-migrator/bin/setenv.sh

```
# Working directory for Migrator

WORKING_DIR=/opt/playce/playce-migrator
```

데이터베이스 환경 구성

Playce RoRo를 사용하기 위해서는 MariaDB 10.6 이상 버전이 대상 서버에 설치되어 있어야 합니다.

참고
데이터베이스는 테이블 및 컬럼의 대소문자 구분을 하지 않습니다.

타임존 설정

DB와 User 생성 전, 타임존을 UTC로 설정하기 위해 /etc/my.cnf 파일을 다음과 같이 수정합니다.

```
# This group is read both both by the client and the server
 # use it for options that affect everything
 #
 [client-server]


 #
 # include all files from the config directory
 #
 !includedir /etc/my.cnf.d


 [mariadb]
 lower_case_table_names = 1
 default_time_zone = '+9:00'
 #max_allowed_packet = 128M
```

참고
Playce RoRo for Java App 로그에 Could not send query: query size is >= to max_allowed_packet 라는 오류 메시지가 발생하면, max_allowed_packet 사이즈를 설정된 용량을 조정한 후 DB 서버를 다시 구동합니다.
max_allowed_packet의 기본 설정 값은 #max_allowed_packet = 16M 입니다.

Database 및 User 생성

데이터베이스와 사용자 생성을 위해 다음의 내용을 실행합니다.

1. Playce RoRo를 위한 데이터베이스와 사용자를 생성합니다.

```
]$ mysql -u root -p

mysql> CREATE DATABASE rorodb CHARACTER SET utf8 COLLATE utf8_bin;

mysql> GRANT ALL PRIVILEGES ON rorodb.* TO playce@'%' IDENTIFIED BY '<PASSWORD>';

mysql> GRANT ALL PRIVILEGES ON rorodb.* TO playce@'localhost' IDENTIFIED BY '<PASSWORD>';

mysql> FLUSH PRIVILEGES;
```

2. Playce RoRo for Java App을 위한 데이터베이스와 사용자를 생성합니다.

```
]$ mysql -u root -p

mysql> CREATE DATABASE migratordb CHARACTER SET utf8 COLLATE utf8_bin;

mysql> GRANT ALL PRIVILEGES ON migratordb.* TO playce@'%' IDENTIFIED BY '<PASSWORD>';

mysql> GRANT ALL PRIVILEGES ON migratordb.* TO playce@'localhost' IDENTIFIED BY '<PASSWORD>';

mysql> FLUSH PRIVILEGES;
```

3. Playce RoRo의 사용자 인증을 위한 oauthdb 데이터베이스와 사용자 권한을 추가합니다.

```
]$ mysql -u root -p

mysql> CREATE DATABASE oauthdb CHARACTER SET utf8 COLLATE utf8_bin;

mysql> GRANT ALL PRIVILEGES ON oauthdb.* TO playce@'%' IDENTIFIED BY '<PASSWORD>';

mysql> GRANT ALL PRIVILEGES ON oauthdb.* TO playce@'localhost' IDENTIFIED BY '<PASSWORD>';

mysql> FLUSH PRIVILEGES;
```

Playce RoRo 패치

Playce RoRo는 주기적인 개선 작업을 통해 버전을 업그레이드하고 있습니다. 새로 릴리즈된 최신 버전으로 업그레이드를 진행하고 싶은 경우, 패치 작업을 통해 최신 버전을 설치할 수 있습니다.

기존 버전의 Playce RoRo에 다운로드한 패치 파일을 적용하면, 내부적으로 프로세스 종료, 기존 설치 디렉터리 백업, 환경 설정 파일 병합, 데이터베이스 업데이트 등의 작업이 자동으로 진행됩니다.

패치 파일 다운로드

패치 작업을 수행하기 전에 다음의 경로에서 패치 파일을 다운로드해야 합니다.

다운로드 위치: https://roro-repository.s3.ap-northeast-2.amazonaws.com/release/playce-patch-1.1.0.tar.gz

> **참고**
>
> 패치 파일 다운로드 위치는 패치 릴리즈 시에 변경됩니다. 주소가 변경될 경우, 릴리즈 시에 공지할 예정입니다.

3.5.1 버전에서 3.6.0 으로 패치하기

기존에 사용 중인 3.5.1 버전에서 3.6.0 버전으로 패치하는 방법은 다음과 같습니다.

1. 다운로드한 playce-patch-1.1.0.tar.gz 패치 파일의 압축을 해제합니다.

```
tar xvf playce-patch-1.1.0.tar.gz
```

2. playce-patch 디렉터리의 start.sh 파일과 patch-automation.jar 파일을 playce_home 상위 디렉터리로 이동시킵니다.

```
mv playce-patch/start.sh playce-patch/patch-automation.jar ${playce_home}/..
```

3. https://www.play-ce.io/resource.html로 접속 후 Playce RoRo 탭에 있는 Playce RoRo 3.6.0 설치 파일을 다운로드 합니다. 파일 이름은 playce-roro-ext-3.6.0.tar.gz 니다.
4. 다운로드한 playce-roro-ext-3.6.0.tar.gz 파일을 playce_home 상위 디렉터리로 이동시킵니다.

```
mv playce-roro-ext-3.6.0.tar.gz ${playce_home}/..
```

5. start.sh을 실행합니다.

```
sh start.sh
```

패치 작업 Flow

패치 파일을 실행하면 내부적으로 다음과 같은 과정을 거쳐 패치를 진행합니다.

1. 실행 중인 Playce RoRo, Playce RoRo for Java App, OAuth, Gateway 프로세스를 종료시킵니다.
2. 기존의 playce 디렉터리를 playce-bak 디렉터리로 변경합니다.
3. 설치 파일 압축을 풉니다.
4. playce-bak 디렉터리의 setenv.sh과 env.sh에 주석 처리되지 않은 변수를 playce 디렉터리의 sh 파일에 병합합니다.
5. 이전 버전과 설치할 최신 버전을 확인하고 업데이트에 필요한 SQL 파일을 찾습니다.
6. SQL 파일의 SQL 쿼리문을 세미콜론(;) 기준으로 구분하여 1개씩 업데이트합니다.

참고

패치 과정 중 문제가 발생해도 롤백 하지 않으므로 먼저 발견된 문제를 해결해야 합니다. 발견된 문제는 데이터베이스 변경 등의 변경이 있는 경우 발생할 수 있습니다. 해당 문제에 따라 조치 방법이 다르므로 자체적으로 문제를 해결하지 못할 경우, 당사 기술지원팀에 연락하시기 바랍니다.
발견된 문제를 해결한 다음에는 playce 디렉터리를 삭제하고 playce-bak 디렉터리를 playce로 이름을 변경한 후 패치를 다시 진행해야 합니다.

Playce Gateway 변경 사항

로드밸런서 등을 사용하는 경우 403 Forbidden 오류가 발생하여 로그인이 되지 않거나 CORS 오류가 발생하는 경우에는 playce-gateway/env.sh 파일의 설정 내용을 일부 변경해야 합니다.

1. {playce_home}/modules/playce-gateway/env.sh 파일을 실행합니다.
2. env.sh 파일에서 다음과 같이 정의된 영역을 찾습니다.

```
#JAVA_OPTS="$JAVA_OPTS-Dspring.cloud.gateway.globalcors.cors-configurations.[/**].allowedOrigins=http://localhost"
```

3. 해당 내용에서 주석(#)을 제거하고 http://localhost에 사용 중인 프로토콜과 도메인을 입력합니다.

```
JAVA_OPTS="$JAVA_OPTS-Dspring.cloud.gateway.globalcors.cors-configurations.[/**].allowedOrigins=https://19*.***.**.77"
```

```
#!/bin/sh

# Set JAVA_HOME when default java version is not JDK 11
# export JAVA_HOME=/usr/lib/jvm/java-11

BASE_DIR="$(cd "$(dirname "$0")"; pwd)";

SERVER_NAME=playce-gateway

PORT_OFFSET=0

HTTP_PORT=$(expr 8080 + $PORT_OFFSET)

OAUTH_OFFSET=`cat $BASE_DIR/../playce-oauth/bin/setenv.sh | egrep '^PORT_OFFSET=' | awk -F '=' '{print $2}'`
RORO_OFFSET=`cat $BASE_DIR/../../playce-roro/bin/setenv.sh | egrep '^PORT_OFFSET=' | awk -F '=' '{print $2}'`
MIGRATOR_OFFSET=`cat $BASE_DIR/../../playce-migrator/bin/setenv.sh | egrep '^PORT_OFFSET=' | awk -F '=' '{print $2}'`

# set route
OAUTH_URL='http://localhost:'$(expr 8080 + $OAUTH_OFFSET)
RORO_URL='http://localhost:'$(expr 8080 + $RORO_OFFSET)
MIGRATOR_URL='http://localhost:'$(expr 8080 + $MIGRATOR_OFFSET)

JAVA_OPTS="-server"
JAVA_OPTS="$JAVA_OPTS -Dserver=$SERVER_NAME"
JAVA_OPTS="$JAVA_OPTS -Dserver.port=$HTTP_PORT"

JAVA_OPTS="$JAVA_OPTS -Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -Djava.security.egd=file:/dev/urandom"
JAVA_OPTS="$JAVA_OPTS -Dsun.rmi.dgc.client.gcInterval=3600000"
JAVA_OPTS="$JAVA_OPTS -Dsun.rmi.dgc.server.gcInterval=3600000"
JAVA_OPTS="$JAVA_OPTS -Dsun.lang.ClassLoader.allowArraySyntax=true"
JAVA_OPTS="$JAVA_OPTS -Djava.awt.headless=true"
JAVA_OPTS="$JAVA_OPTS -Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -Dserver.info=real"
JAVA_OPTS="$JAVA_OPTS -Dfile.encoding=UTF-8 -Dfile.client.encoding=UTF-8"

# application config
if [ -n "$OAUTH_URL" ]; then
  echo 'route oauth: '$OAUTH_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[0].id=oauth"
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[0].uri="$OAUTH_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[0].predicates=Path=/oauth/**"
fi
if [ -n "$RORO_URL" ]; then
  echo 'route roro: '$RORO_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[1].id=roro"
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[1].uri="$RORO_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[1].predicates=Path=/roro/**"
fi
if [ -n "$MIGRATOR_URL" ]; then
  echo 'route migrator: '$MIGRATOR_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[2].id=migrator"
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[2].uri="$MIGRATOR_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[2].predicates=Path=/migrator/**"
fi

# JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.globalcors.cors-configurations.[/**].allowedOrigins=http://localhost"
export JAVA_OPTS
```

[그림 1] 수정 전 env.sh 파일 내용

```
#!/bin/sh

# Set JAVA_HOME when default java version is not JDK 11
# export JAVA_HOME=/usr/lib/jvm/java-11

BASE_DIR="$(cd "$(dirname "$0")"; pwd)";

SERVER_NAME=playce-gateway

PORT_OFFSET=0

HTTP_PORT=$(expr 8080 + $PORT_OFFSET)

OAUTH_OFFSET=`cat $BASE_DIR/../playce-oauth/bin/setenv.sh | egrep '^PORT_OFFSET=' | awk -F '=' '{print $2}'`
RORO_OFFSET=`cat $BASE_DIR/../../playce-roro/bin/setenv.sh | egrep '^PORT_OFFSET=' | awk -F '=' '{print $2}'`
MIGRATOR_OFFSET=`cat $BASE_DIR/../../playce-migrator/bin/setenv.sh | egrep '^PORT_OFFSET=' | awk -F '=' '{print $2}'`

# set route
OAUTH_URL='http://localhost:'$(expr 8080 + $OAUTH_OFFSET)
RORO_URL='http://localhost:'$(expr 8080 + $RORO_OFFSET)
MIGRATOR_URL='http://localhost:'$(expr 8080 + $MIGRATOR_OFFSET)

JAVA_OPTS="-server"
JAVA_OPTS="$JAVA_OPTS -Dserver=$SERVER_NAME"
JAVA_OPTS="$JAVA_OPTS -Dserver.port=$HTTP_PORT"

JAVA_OPTS="$JAVA_OPTS -Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -Djava.security.egd=file:/dev/urandom"
JAVA_OPTS="$JAVA_OPTS -Dsun.rmi.dgc.client.gcInterval=3600000"
JAVA_OPTS="$JAVA_OPTS -Dsun.rmi.dgc.server.gcInterval=3600000"
JAVA_OPTS="$JAVA_OPTS -Dsun.lang.ClassLoader.allowArraySyntax=true"
JAVA_OPTS="$JAVA_OPTS -Djava.awt.headless=true"
JAVA_OPTS="$JAVA_OPTS -Djava.net.preferIPv4Stack=true"
JAVA_OPTS="$JAVA_OPTS -Dserver.info=real"
JAVA_OPTS="$JAVA_OPTS -Dfile.encoding=UTF-8 -Dfile.client.encoding=UTF-8"

# application config
if [ -n "$OAUTH_URL" ]; then
  echo 'route oauth: '$OAUTH_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[0].id=oauth"
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[0].uri="$OAUTH_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[0].predicates=Path=/oauth/**"
fi
if [ -n "$RORO_URL" ]; then
  echo 'route roro: '$RORO_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[1].id=roro"
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[1].uri="$RORO_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[1].predicates=Path=/roro/**"
fi
if [ -n "$MIGRATOR_URL" ]; then
  echo 'route migrator: '$MIGRATOR_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[2].id=migrator"
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[2].uri="$MIGRATOR_URL
  JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.routes[2].predicates=Path=/migrator/**"
fi

JAVA_OPTS="$JAVA_OPTS -Dspring.cloud.gateway.globalcors.cors-configurations.[/**].allowedOrigins=https://19      .77"
export JAVA_OPTS
```

[그림 2] 수정 후 env.sh 파일 내용

3 장 실행하기

서비스 실행

Playce RoRo 통합 버전의 설치 및 설정을 마친 후 정상 작동 여부를 확인합니다.

Java 버전 확인

서비스를 실행하기 전 서버에 설치된 Java 버전을 확인합니다.

```
]$ java -version

java 11.0.13
```

Java 설치

java 명령을 찾을 수 없거나 설치된 버전이 11.0.13 보다 낮은 경우, 다음과 같이 Java 패키지를 설치하고 설정합니다.

> **참고**
> JRE 설치 시 일부 기능 사용에 제한이 발생할 수 있으므로 JDK 설치를 권장합니다.

1. Java 패키지를 설치합니다.

```
]$ sudo yum install -y java-11-openjdk-devel
```

2. ~/.bash_profile 파일에 JAVA_HOME 및 PATH 환경변수를 다음과 같이 설정합니다.

```
# ~/.bash_profile을 수정합니다


vi ~/.bash_profile

...


# JDK 설치 경로는 버전 및 시스템에 따라 다를 수 있습니다.

# 기존 PATH에 $JAVA_HOME/bin이 포함될 수 있도록 아래와 같이 추가합니다.

export JAVA_HOME=/usr/lib/jvm/java-11

PATH=$JAVA_HOME/bin:$PATH:$HOME/bin
```

설치 확인

Playce RoRo 설치 및 설정을 마친 후 정상 작동 여부를 확인합니다.

1. Playce 제품 홈 디렉터리로 이동합니다. /opt에 제품을 설치한 경우 디렉터리의 이동은 다음과 같습니다.

```
[roro@localhost ~]$ cd /opt/playce
```

2. bin/start-all.sh 스크립트를 실행합니다. 스크립트를 실행하면 총 4개의 Java 프로세스가 실행됩니다.

```
[roro@localhost /opt/playce]$ sh bin/start-all.sh
```

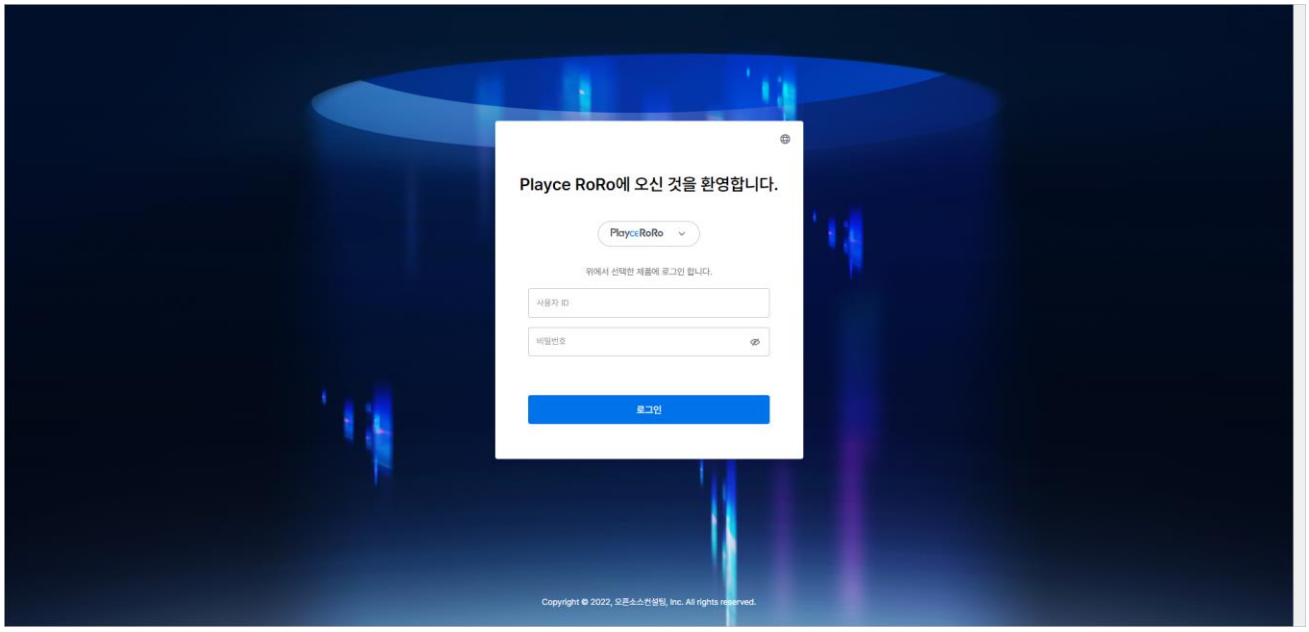
3. Playce RoRo 개별 서버의 포트인 8080, 8081, 8082, 8083번 포트가 정상적으로 통신 가능한지 확인하기 위해 다음의 명령을 실행합니다.

```
[roro@localhost /opt/playce]$ netstat -anp | grep <SERVICE_PORT>
```

4. 웹 브라우저 실행 후 http://<server-ip>:${playce_gateway_port}에 접속하여 정상 접속 여부를 확인합니다. ${playce_gateway_port}에는 기본 포트에 설치한 경우에는 게이트웨이 기본 포트인 8080을 입력하고, 포트를 변경한 경우에는 변경한 게이트웨이 포트를 입력합니다.

> **참고**
>
> ${PLAYCE_HOME}/bin/start-all.sh로 모든 서버를 한 번에 실행할 수도 있으며, 각 서버 모듈별 디렉터리로 접속하여 bin 디렉터리 하위의 시작 스크립트로 실행할 수 있습니다.

5. Playce RoRo 통합 로그인 화면이 표시되는지 확인합니다.



방화벽 설정 수정

시스템에 설치된 방화벽이 active 상태로 브라우저 접속이 불가능한 경우 다음 명령어를 실행하여 게이트웨이 수신 포트의 접속을 허용합니다.

```
$ sudo firewall-cmd --add-port=8080/tcp --permanent

$ sudo firewall-cmd --add-port=8080/tcp
```

4 장 시작하기

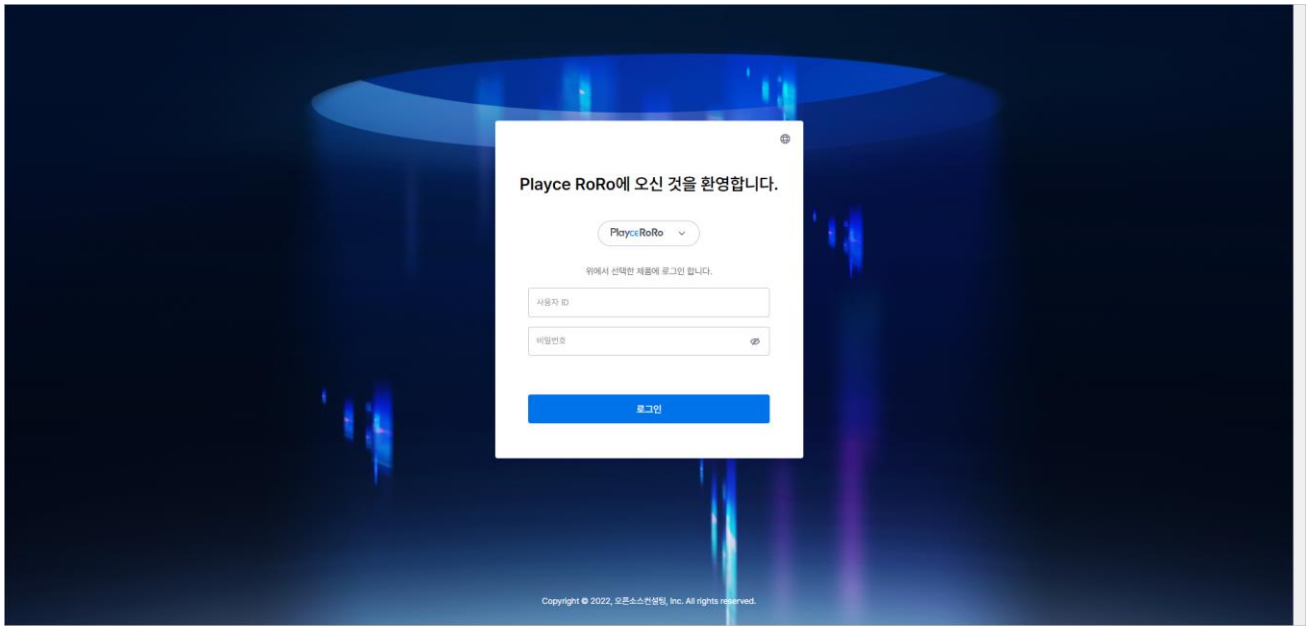
로그인

Playce RoRo에 로그인하려면 웹 브라우저 실행 후 사용자 ID와 비밀번호를 입력해야 합니다.

참고
사용자 ID가 없는 경우, Playce RoRo 관리자에게 요청하여 사용자 ID와 초기 비밀번호를 발급받으십시오.

참고
웹 브라우저는 Chrome 버전 47.x 이상을 권장합니다.

1. 웹 브라우저의 주소 입력 창에 Playce RoRo 로그인 서버 주소를 입력합니다.
2. 로그인 화면에서 로그인할 대상 제품으로 **Playce RoRo**를 선택합니다.
3. **사용자 ID**와 **비밀번호**를 입력한 후 **로그인**을 클릭합니다.



- 언어(): 콘솔 화면에서 사용할 언어를 선택합니다. 초기 설정 언어는 브라우저에 설정된 언어에 따라 언어 선택에 따라 자동 설정됩니다. 선택할 수 있는 언어는 영어(English)와 한글(Korean)입니다.
- 제품 선택: 로그인할 대상 제품을 선택합니다. Playce RoRo와 Playce RoRo for Java App을 선택할 수 있습니다.
- 사용자 ID: 사용자 ID를 입력합니다.
- 비밀번호: 관리자에게 받은 초기 비밀번호로 로그인 한 후, 비밀번호를 변경해야 합니다. 새 비밀번호는 현재 비밀번호와 다르게 설정해야 합니다.
- 비밀번호 숨김/표시(/): 입력한 비밀번호를 ***로 숨겨서 표시하거나 입력된 문자열을 그대로 표시합니다.

참고

사용자 ID나 비밀번호를 잘못 입력한 경우, **사용자 ID 또는 비밀번호를 잘못 입력했습니다. 입력한 내용을 다시 확인하십시오.(로그인을 7번 실패할 경우 계정이 잠깁니다.)** 라는 메시지가 표시됩니다.

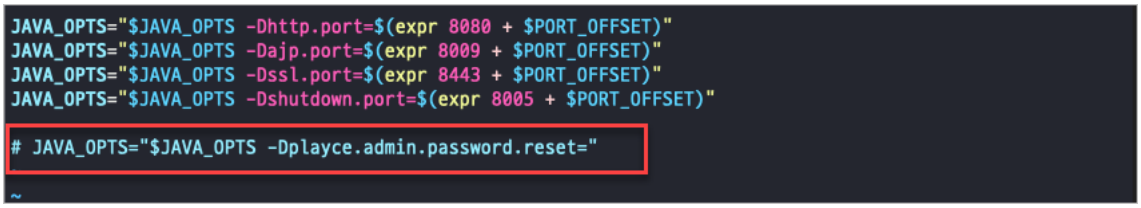
비밀번호 초기화

사용자가 비밀번호를 분실했거나 로그인 시도 횟수 7회를 초과하여 계정이 잠긴 경우에는 비밀번호를 초기화해야 합니다.

1. Playce 제품 홈 디렉터리로 이동하여 다음의 경로에 접속합니다.

```
${playce_home}/modules/playce-oauth/bin
```

2. vi 명령으로 setenv.sh 파일을 엽니다.



3. setenv.sh 파일에서 주석 처리되어 있는 비밀번호 Reset 명령어 라인에서 주석(#)을 제거하고 변경할 비밀번호 문자열을 입력합니다.

```
JAVA_OPTS="$JAVA_OPTS -Dplayce.admin.password.reset=변경할 비밀번호"
```

4. oauth 서버를 다시 시작합니다.

```
sh ${Playce_home}/modules/playce-oauth/bin/kill.sh

sh ${Playce_home}/modules/playce-oauth/bin/startup.sh
```

참고

비밀번호 초기화 이후에는 서버를 반드시 다시 시작해야 합니다.

5. setenv.sh 파일을 다시 실행한 후 비밀번호 Reset 명령어 라인을 2번 단계의 그림과 같이 주석 처리합니다.

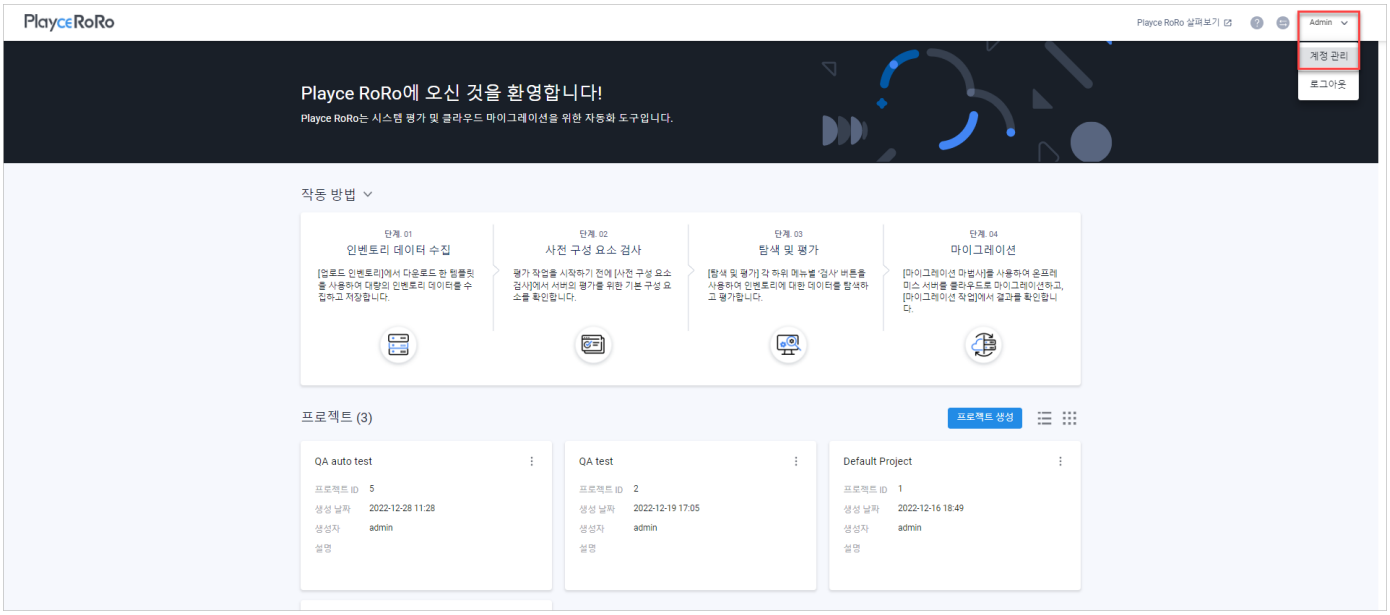
```
# JAVA_OPTS="$JAVA_OPTS -Dplayce.admin.password.reset="
```

비밀번호 변경

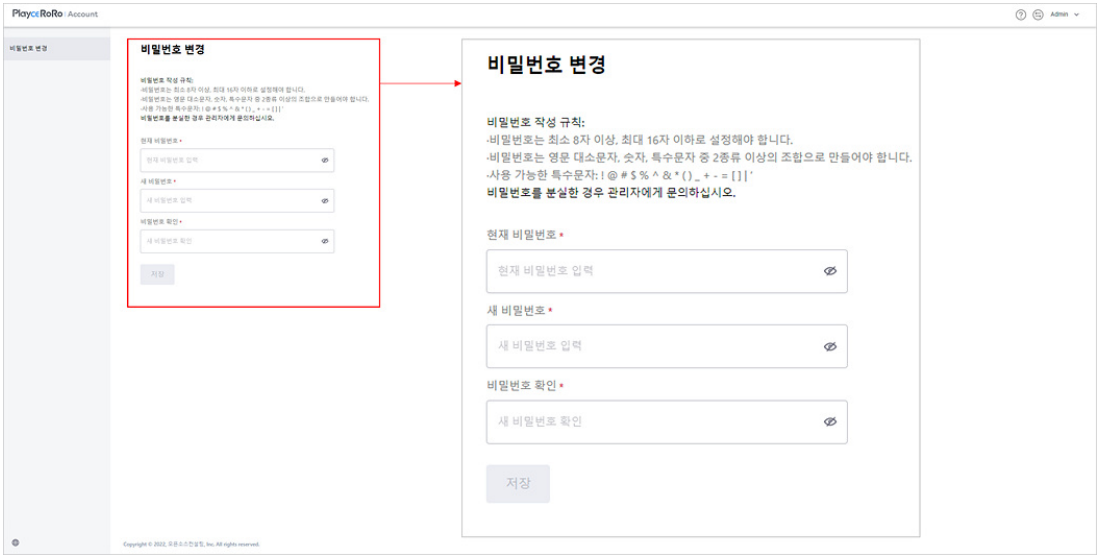
관리자에게 받은 초기 비밀번호나 사용 중인 비밀번호를 변경할 수 있습니다.

> **참고**
> 초기 비밀번호를 로그인 후 반드시 변경하여 사용하시기 바랍니다.

1. 웹 브라우저의 주소 입력 창에 Playce RoRo 로그인 서버 주소를 입력합니다.
2. 로그인 화면에서 로그인할 대상 제품을 선택하고 **사용자 ID**와 **비밀번호**를 입력한 후 **로그인**을 클릭합니다.
3. 화면 오른쪽 위에 표시되는 사용자 ID를 클릭하면 표시되는 **계정 관리**를 클릭합니다.



4. <비밀번호 변경>이 나타납니다. **현재 비밀번호**와 **새 비밀번호**를 설정합니다.



- 현재 비밀번호: 현재 사용 중인 비밀번호를 입력합니다.
- 새 비밀번호: 새로 설정할 비밀번호를 작성 규칙에 따라 입력합니다.
- 비밀번호 확인: 새 비밀번호에 입력한 내용과 동일하게 입력합니다.

5. 입력한 내용을 저장하려면 **저장**을 클릭합니다.

비밀번호 작성 규칙

비밀번호는 다음의 규칙에 따라 만들어야 합니다.

- 비밀번호 길이: 최소 8자 이상, 최대 16자 이하
- 비밀번호 조합 규칙: 영문 대소문자, 숫자, 특수문자 중 2종류 이상의 조합
- 사용 가능한 특수문자: ! @ # $ % ^ & * ( ) _ + - = [ ] | ’

오픈소스컨설팅
OPEN SOURCE CONSULTING

서울특별시 강남구 테헤란로83길 32 나라키움삼성동 A빌딩 2, 5층 (우)06167
Tel.02-516-0711 | **Fax.**02-516-0722 | https://www.osci.kr
Copyright © Open Source Consulting, Inc. All rights reserved.