

# IC3000에서 실행 중인 애플리케이션/컨테이너의 콘솔에 액세스하는 방법

## 목차

### [소개](#)

[IC3000에서 실행 중인 애플리케이션/컨테이너의 콘솔에 액세스하는 방법](#)

[IC3000을 개발자 모드로 설정](#)

[IOx 애플리케이션 구축, 활성화 및 시작](#)

## 소개

이 문서에서는 문제 해결 및 디버깅을 위해 IC3000에서 실행 중인 애플리케이션/컨테이너의 콘솔에 액세스하는 방법에 대해 설명합니다.

## IC3000에서 실행 중인 애플리케이션/컨테이너의 콘솔에 액세스하는 방법

IC3000에서 콘솔에 액세스하려면 IC3000이 개발자 모드에서 실행 중이어야 합니다. IC3000이 프로덕션 모드에 있으면 애플리케이션 콘솔에 액세스할 수 없습니다. 이것은 설계에 의한 것입니다.

### IC3000을 개발자 모드로 설정

실행 중인(또는 실패) 애플리케이션을 디버그/트러블슈팅하려면 먼저 IC3000이 개발자 모드인지 확인합니다.

개발자 모드에서는 IC3000이 관리되지 않는 개발 디바이스이며 로컬 관리자 및 iox 클라이언트를 통해 제어됩니다.

1. USB 또는 직렬 콘솔 케이블을 사용하여 콘솔에 연결합니다.
2. "developer" 사용자의 암호를 설정합니다(명령 개발자 set-password 사용).

```
ic3k>developer set-password
Enter password:
Re-enter password:
Password set successfully!
```

3. 네트워크 케이블을 사용하여 IC3000의 관리 인터페이스를 컴퓨터에 연결합니다.
4. 컴퓨터의 네트워크 인터페이스에 "169.254.128.4(netmask 255.255.0.0)" IP 주소를 할당합니다. 이 특정 IPv4 링크-로컬 주소를 할당해야 합니다.
5. IC3000은 30초 내에 개발자 모드에서 작동할 준비가 됩니다(30초의 지연은 IC3000이 처음 부팅될 때만 발생합니다. 이후의 모든 리로드는 즉시 IC3000을 개발자 모드로 전환합니다(지체 없이).
6. 다음 명령을 사용하여 IC3000이 개발자 모드인지 확인할 수 있습니다.

```
ic3k>show ida status
Status: Running
Operation Mode: Developer
FND Host: N/A
FND Connection Status: Not Connected
Periodic Metrics Interval: N/A
Heartbeat Interval: N/A
Is Registered: False
HTTP Server Status: Running
```

7. 컴퓨터의 브라우저에서 "<https://169.254.128.2:8443>"에 액세스합니다.
8. 2단계에서 설정한 "개발자" 사용자 및 비밀번호를 사용하여 로그인합니다.

## IOx 애플리케이션 구축, 활성화 및 시작

IOx 애플리케이션 패키지를 구축, 활성화 및 시작하려면 브라우저를 통해 iox 클라이언트 또는 로컬 관리자를 사용합니다.

선택적으로, 응용 프로그램을 활성화할 때 디버그 옵션을 활성화할 수 있습니다.이렇게 하면 이미지에 표시된 것처럼 응용 프로그램이 예기치 않게 종료될 때 애플리케이션 컨테이너가 중지되지 않습니다.

▼ Resources

▼ Resource Profile

Profile:

CPU  cpu-units

Memory  MB

Disk  MB

Avail. CPU (cpu-units) 10260 Avail. Memory (Mb) 6400

Activate  debug mode (For troubleshooting only)

응용 프로그램 콘솔에 연결합니다.

응용 프로그램이 실행되거나 디버그 모드에 있으면 다음과 같이 컨테이너의 콘솔에 연결할 수 있습니다.

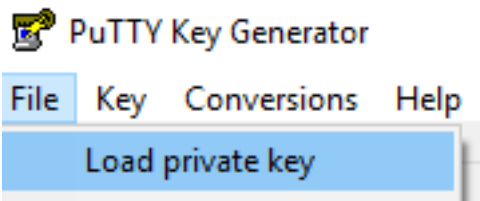
먼저 로컬 관리자 또는 ioxclient를 사용하여 컨테이너의 개인 키를 다운로드합니다. Local Manager(로컬 관리자) > Applications(애플리케이션) > <특정 앱> > Manage(관리) > App Info(앱 정보) > \*.pem 파일(이미지에 표시됨)

App Access	
Console Access	ssh -p {SSH_PORT} -i test.pem appconsole@169.254.128.2

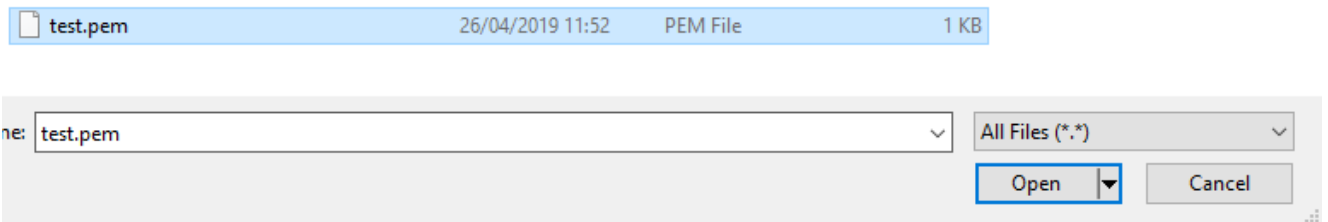
Linux에서 SSH-port 22를 사용하여 나열된 명령을 실행합니다. 예를 들면 다음과 같습니다. ssh -i test.pem appconsole@169.254.128.2

Windows에서 Putty를 사용하는 경우 먼저 PuTygen을 사용하여 .pem 파일을 Putty 호환 .ppk로 변환합니다.

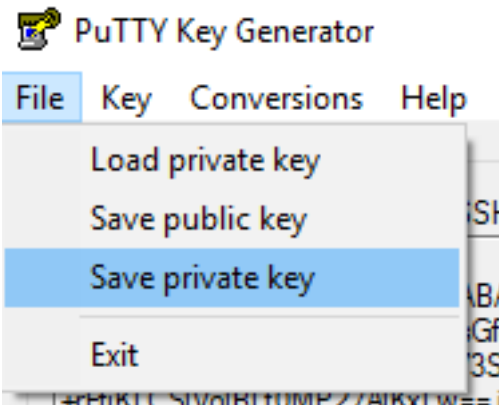
1. PuTygen 시작
2. 이미지에 표시된 대로 **File(파일) > Load private key(개인 키 로드)**로 이동합니다.



3. 파일 필터를 **All Files(모든 파일)**로 설정하고 이미지에 표시된 대로 다운로드한 .pem을 엽니다.

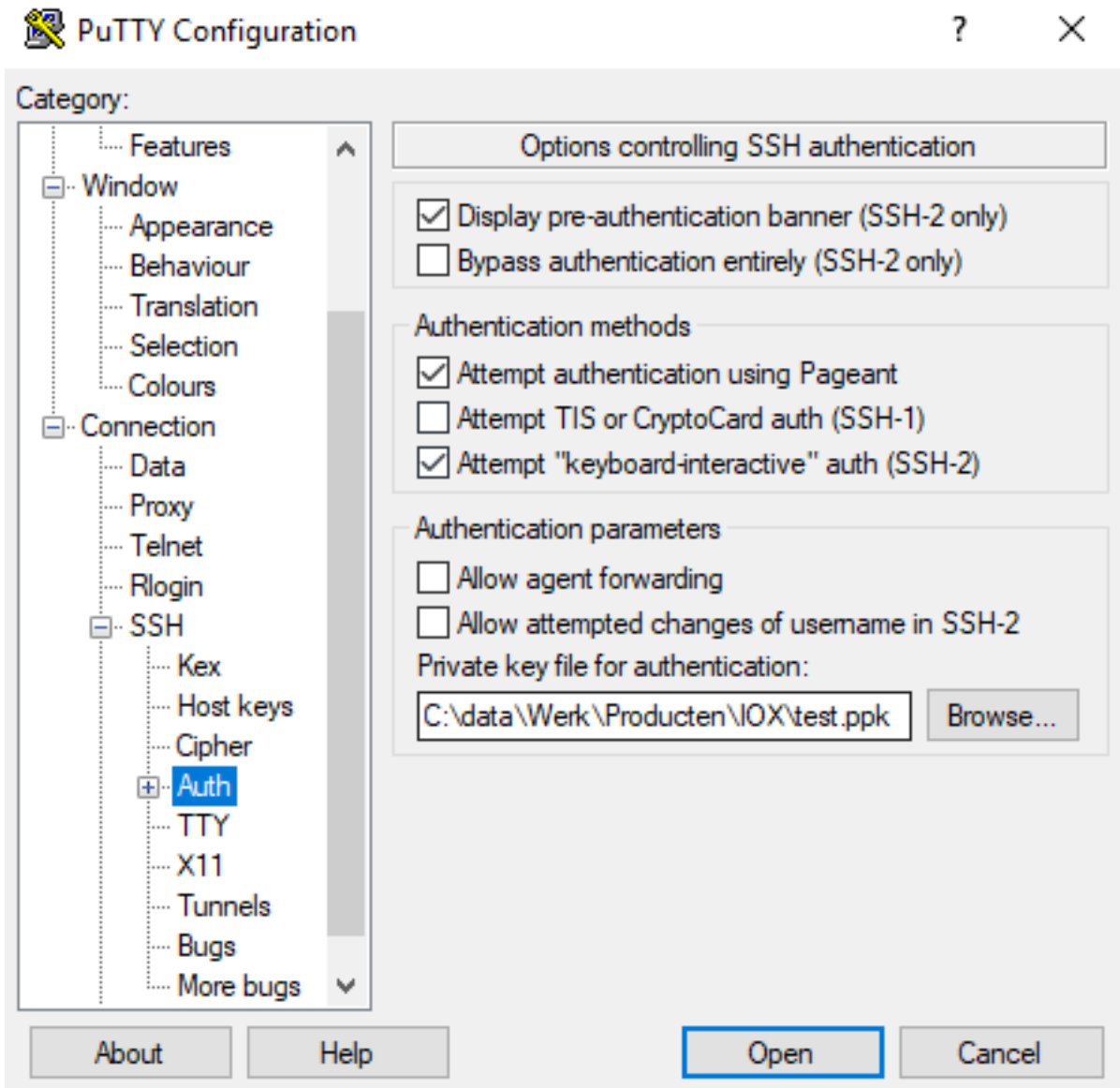


4. **File(파일) > Save private key(개인 키 저장)**로 이동하고 이미지에 표시된 대로 .pem을 .ppk로 저장합니다.

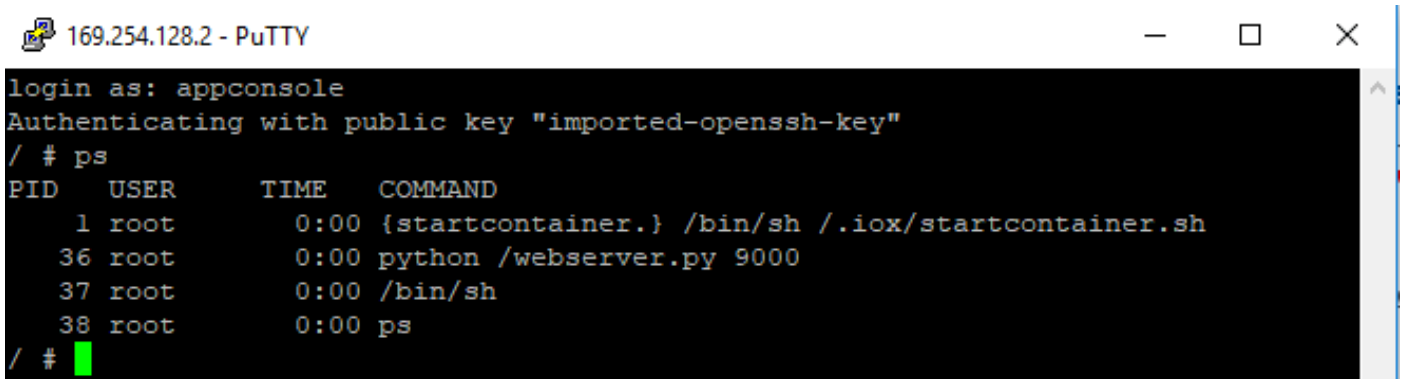


.ppk가 있는 경우 PuTTY를 시작하고 세션 대화 상자에 169.254.128.2 포트 22를 입력합니다.

다음으로, **Connection - SSH - Auth**로 이동하고 이미지에 표시된 대로 .ppk 파일을 제공합니다.



세션을 시작하려면 열기를 클릭합니다. 이미지에 표시된 대로 사용자 이름으로 "appconsole"을 입력합니다.



그러면 IC3000에서 실행 중인 IOx 컨테이너의 애플리케이션 콘솔로 연결됩니다.