
OpenShift에 JENNIFER 설치하기

1. JAVA Application이 구동되는 OpenShift 환경



Red Hat 배포판 OpenShift의 기본적인 사용법은 알고 있다고 가정하고 작성되었습니다.

이번 예제는, Red Hat OpenShift V3에서 Java Application을 구동하는 것을 JENNIFER java 에이전트로 모니터링하는 튜토리얼입니다.

1.1 사전 준비

- [Source to Image \(S2I\)](https://github.com/openshift/source-to-image) 웹사이트 : <https://github.com/openshift/source-to-image>
- [OpenShift CLI OC](#)



이 튜토리얼에서는 별도로 구동되고 있으며, 유효한 라이선스가 있는 JENNIFER Server가 존재합니다. Java Agent 패키지와 JENNIFER Server가 없다면, 홈페이지에서 2주짜리 무료 트라이얼 라이선스를 신청하여 설치하면 됩니다.

1.2 요약

- JENNIFER Java agent 패키지 다운로드
- 해당 패키지가 포함된 JENNIFER agent builder image 제작
- builder image를 이용해 application 이미지를 제작
- OpenShift에 배포

1.3 JENNIFER builder image 제작



이번 튜토리얼은 Wildfly 12 Final 버전을 예로 구성합니다.

1.3.1 Repository Clone하기

아래 링크의 Github Repository를 Clone합니다

<https://github.com/JENNIFER-University/s2i-wildfly-jennifer> 혹은 s2i-wildfly 프로젝트에서 원하는

버전을 찾아 Clone 합니다.

<https://github.com/openshift-s2i/s2i-wildfly>

1.3.2 JENNIFER Java agent 패키지

JENNIFER Java agent 패키지를 다운로드 하여, 해당 프로젝트의 Root 디렉토리에 위치시킵니다.

1.3.3 JENNIFER.conf 파일

에이전트 설정 파일을 수정합니다. /control/jennifer.conf 파일을 편집해 아래와 같은 형태로 만들어 주세요. server_a ddress와 server_port 가 접속하고자 하는 JENNIFER Data server의 IP와 Port, 해당 에이전트가 속할 domain_id를 반드시 지정해 주어야 합니다.

```
server_address = 192.168.9.37
server_port = 5000

# default log_dir "{agent_home}/logs"
# log_dir = {agent_home}/logs
# log_file =
```

```
# log_level : 1(info), 2(debug), 3(trace)
log_level = 1
log_encoding = UTF-8
log_rotation = true
# log_max_age : Working when log_rotation is activated. Valid range is (1-365)
# log_max_age = 0

# domain_id : Valid range is (1-32767)
domain_id = 1000
```



jennifer-agent-java-x.x.x.x.zip 형태의 에이전트 설치 패키지에 압축을 해제해 보면, **jennifer.conf** 파일이 있는데, 해당 파일을 복사하고, 편집해 사용할 수도 있습니다. 다만, 원본 **zip**파일이 **Root** 작업 디렉토리에 존재해야 합니다.

1.3.4 builer image Build 하기

```
docker build -t jennifer-oc-builder .
```

jennifer-oc-builder 라는 이름의 새로운 이미지가 생성되었다면 성공입니다.

1.4 Application image 제작

builder image가 생성되었다면, 애플리케이션 이미지를 생성할 수 있습니다. 여기서는 "**openshift-jee-sample**" 이라고 오픈쉬프트에서 제공하는 이미지를 이용하겠습니다. 아래 링크에서 확인할 수 있습니다.

<https://github.com/openshift/openshift-jee-sample>



application image를 **build**하기 위해서 **s2i** 스크립트를 사용합니다. 설치 및 **PATH** 설정이 잘 되어있는지 확인하세요. 더 자세한 사항은 <https://github.com/openshift/source-to-image> 문서에서 확인할 수 있습니다.

1.4.1 s2i build 스크립트 실행하기.

build 명령은 **s2i build REPO_LINK BUILD_IMAGE APP_IMAGE_NAME** 의 형식을 가집니다.

```
s2i build https://github.com/openshift/openshift-jee-sample jennifer-oc-builder jennifer-oc-
```

app jennifer-oc-app 이라는 Application Image가 생성되면 성공입니다.

1.4.2 image Stream 생성하기

```
oc login OPENSIFT_SERVER_IP  
oc create is jennifer-oc-app -n myproject
```

oc create is 명령어로 jennifer-oc-app이라는 이름을 가진 image stream을 myproject라는 namespace에 생성합니다.

1.4.3 application image를 태그하고, openshift 내부에 등록합니다.

```
docker tag jennifer-oc-app oc-docker-reigstry.dev/myproject/jennifer-oc-app  
docker login oc-docker-reigstry.dev  
docker push oc-docker-reigstry.dev/myproject/jennifer-oc-app
```

1.5 OpenShift에 배포하기

1. OpenShift Console에 로그인 합니다.
2. 위에서 생성한 image stream이 속한 namespace를 선택합니다. 위 예제에서는 myproject 입니다.
3. jennifer-oc-app 이라고 생성한 image stream을 선택하여 배포합니다.
4. JENNIFER 대시보드에 새로운 instance가 접속하는 것을 확인합니다.