

Dev, Back-End/CI & CD

[JenKins] ② 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

TROLL's | 2021. 6. 1. 21:00

②
JenKins를 이용해서
node.js CI/CD
자동 배포하는 방법
(NestJS)

JenKins를 이용해서 node.js 배포하는 방법 (NestJS) 1편에 이어서

2편을 포스팅하겠습니다.

▼ Jenkins를 통해 자동배포 (NodeJS) 시리즈

- ▶ ① [젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기](#)
- ▶ ② [젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기](#)
- ▶ ③ [젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기](#)
- ▶ ④ [젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기](#)
- ▶ ⑤ [젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 - 바로가기](#)



오늘은 이전에 젠킨스 설치했던 AWS EC2서버에 브라우저로 접속하여 관리자 계정 생성을 해주겠습니다.

● AWS 젠킨스 EC2 Security Group 변경

인바운드 규칙 편집 정보

인바운드 규칙은 인스턴스에 도달하도록 허용된 수신 트래픽을 제어합니다.

유형	정보	프로토콜	정보	포트 범위	정보	소스	정보	설명 - 선택 사항	정보
SSH		TCP		22		사용자 ...	Q		삭제
							0.0.0.0/0	X	
사용자 지정 TCP		TCP		8080		위치 무관	Q		삭제
							0.0.0.0/0	X	
							::/0	X	

[규칙 추가](#)

참고: 기존 규칙을 편집하면 편집된 규칙이 삭제되고 새 세부 정보로 새 규칙이 생성됩니다. 이렇게 하면 새 규칙이 생성될 때까지 해당 규칙에 의존하는 트래픽이 잠시 중단될 수 있습니다.

1. 보안그룹에서 인바운드 규칙 편집으로 들어가서 기존에 있던 SSH는 그대로 두고, 8080포트를 위치 무관으로 해서 모두 열어줍니다.

젠킨스 서비스에 액세스 하기 위해서 2개의 인바운드(들어오는) Rule을 설정해준것입니다.

▼ 인스턴스 요약 정보

인스턴스 ID

 [redacted] (ovni_server)

인스턴스 상태

 실행 중

퍼블릭 IPv4 주소

 [redacted] [개방 주소법](#)

퍼블릭 IPv4 DNS

 [redacted]
[redacted] [개방 주소법](#)

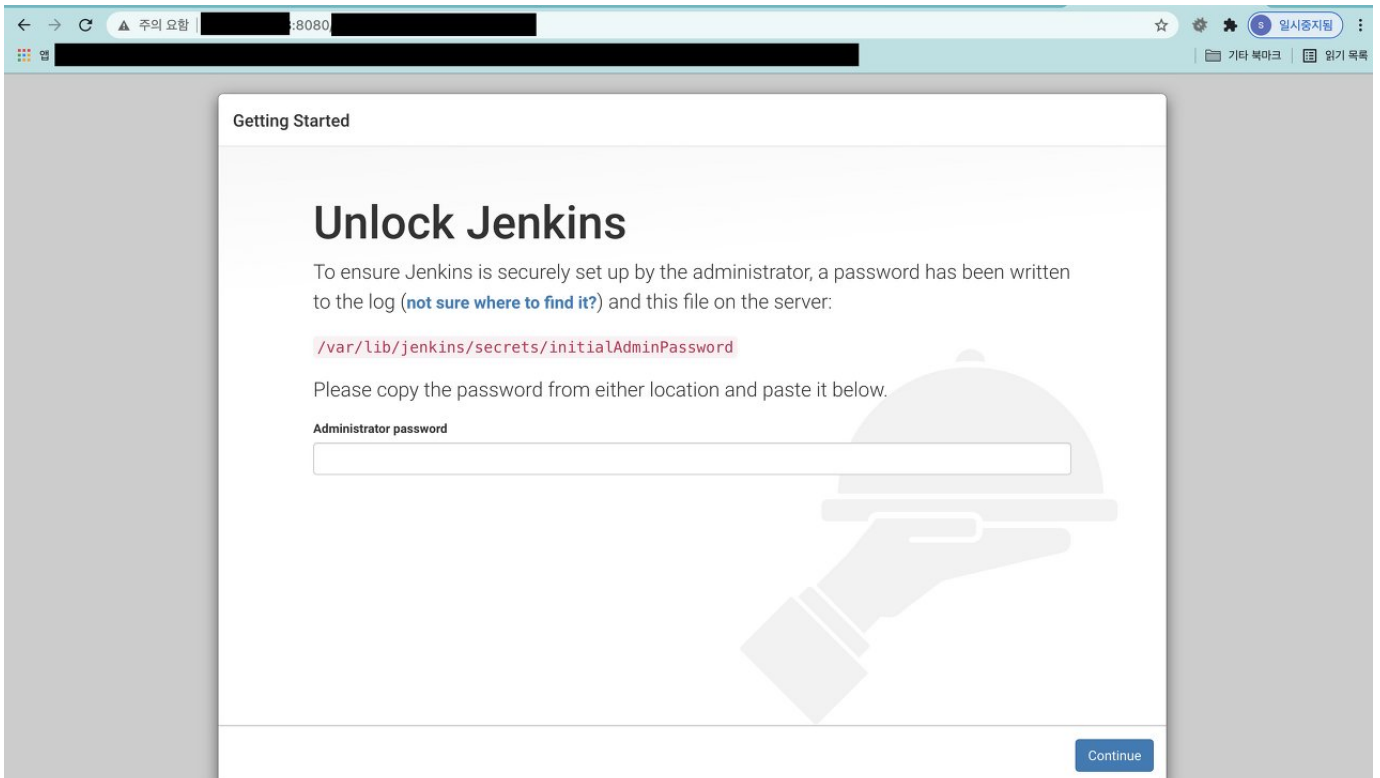
프라이빗 IPv4 주소

 [redacted]

프라이빗 IPv4 DNS

 [redacted]

2. EC2 인스턴스에서 해당 젠킨스 서버를 클릭하여 하단에 퍼블릭 IPv4 주소(빨간 박스 부분)를 복사해줍니다.



3. 위에서 복사한 IP주소에 8080포트를 이어서 붙여 브라우저에 접속해줍니다.

접속이 된다면 정상적으로 Jenkins를 정상적으로 설치한 것 입니다.

http://젠킨스인스턴스IP주소:8080

```
[ubuntu@ip- ~]$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

```
$ sudo cat /var/lib/jenkins/secrets/initialAdminPassword
```

4. 다시 젠킨스 서버에 SSH로 접속하여 시크릿 값 (파란박스 부분)를 가져옵니다.

명령어를 입력하고 나온 값을 그대로 복사해줍니다.

Getting Started

Unlock Jenkins

To ensure Jenkins is securely set up by the administrator, a password has been written to the log (**not sure where to find it?**) and this file on the server:

`/var/lib/jenkins/secrets/initialAdminPassword`

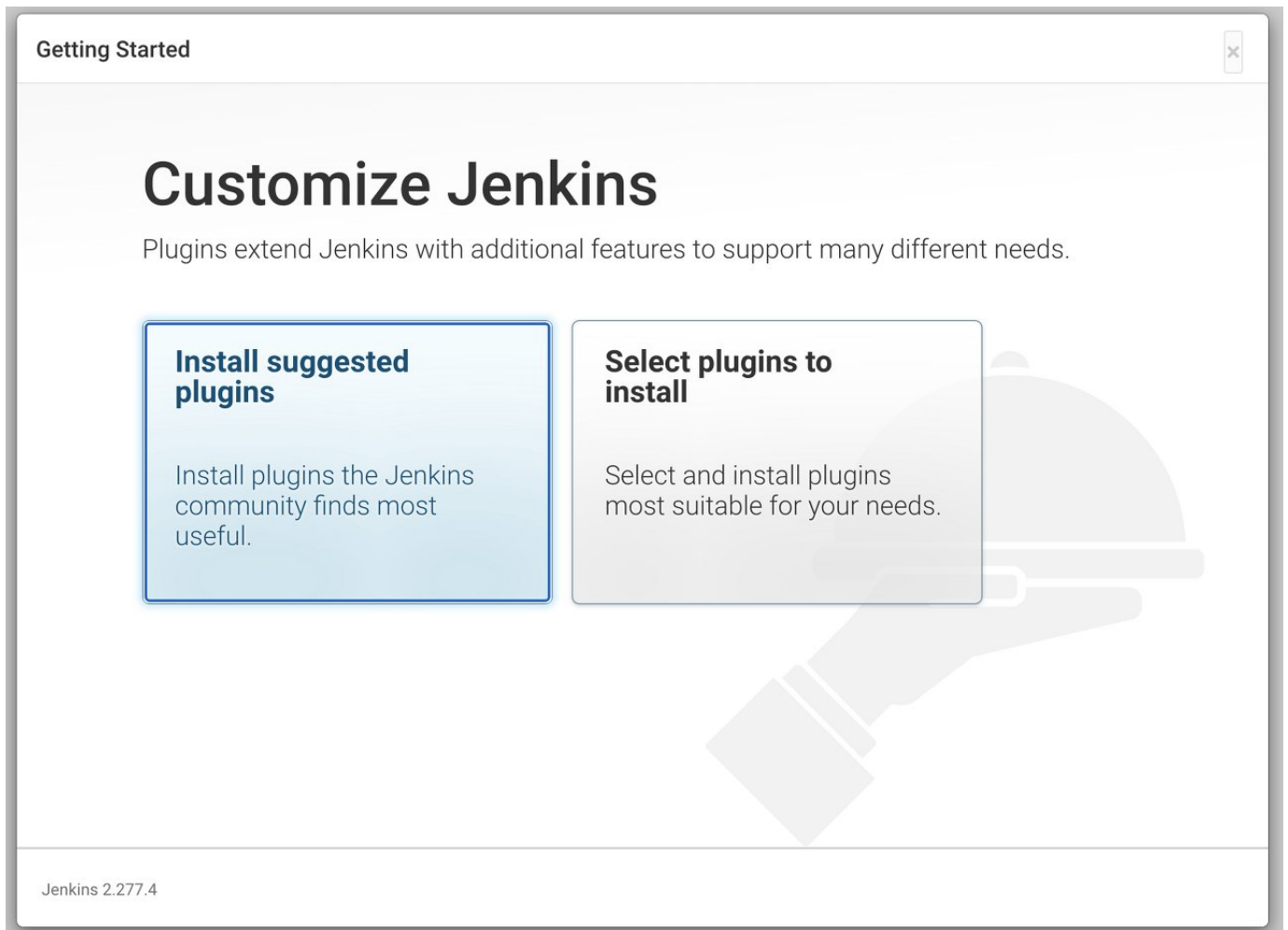
Please copy the password from either location and paste it below.

Administrator password

.....|

Continue

5. 다시 브라우저로 젠킨스 서버에 접속해줍니다. 4번에서 복사한 시크릿 값을 붙여넣기 해줍니다.



6. Install Suggested plugins을 클릭해줍니다.

Getting Started

✓ Folders	OWASP Markup Formatter	Build Timeout	Credentials Binding	Folders
Timestampers	Workspace Cleanup	Ant	Gradle	** Trilead API
Pipeline	GitHub Branch Source	Pipeline: GitHub Groovy Libraries	Pipeline: Stage View	OWASP Markup Formatter
Git	SSH Build Agents	Matrix Authorization Strategy	PAM Authentication	
LDAP	Email Extension	Mailer		

Jenkins 2.277.4

** - required dependency

7. 위와 같이 뜬다면 설치중이므로, 모두 설치될때까지 기다려줍니다.

Create First Admin User

계정명:

암호:

암호 확인:

이름:

이메일 주소:

8. 계정명부터 이메일주소까지 모두 입력한뒤에 저장하고 계속 버튼을 클릭합니다.
계정명과 암호는 반드시 기억해줍니다.

Getting Started

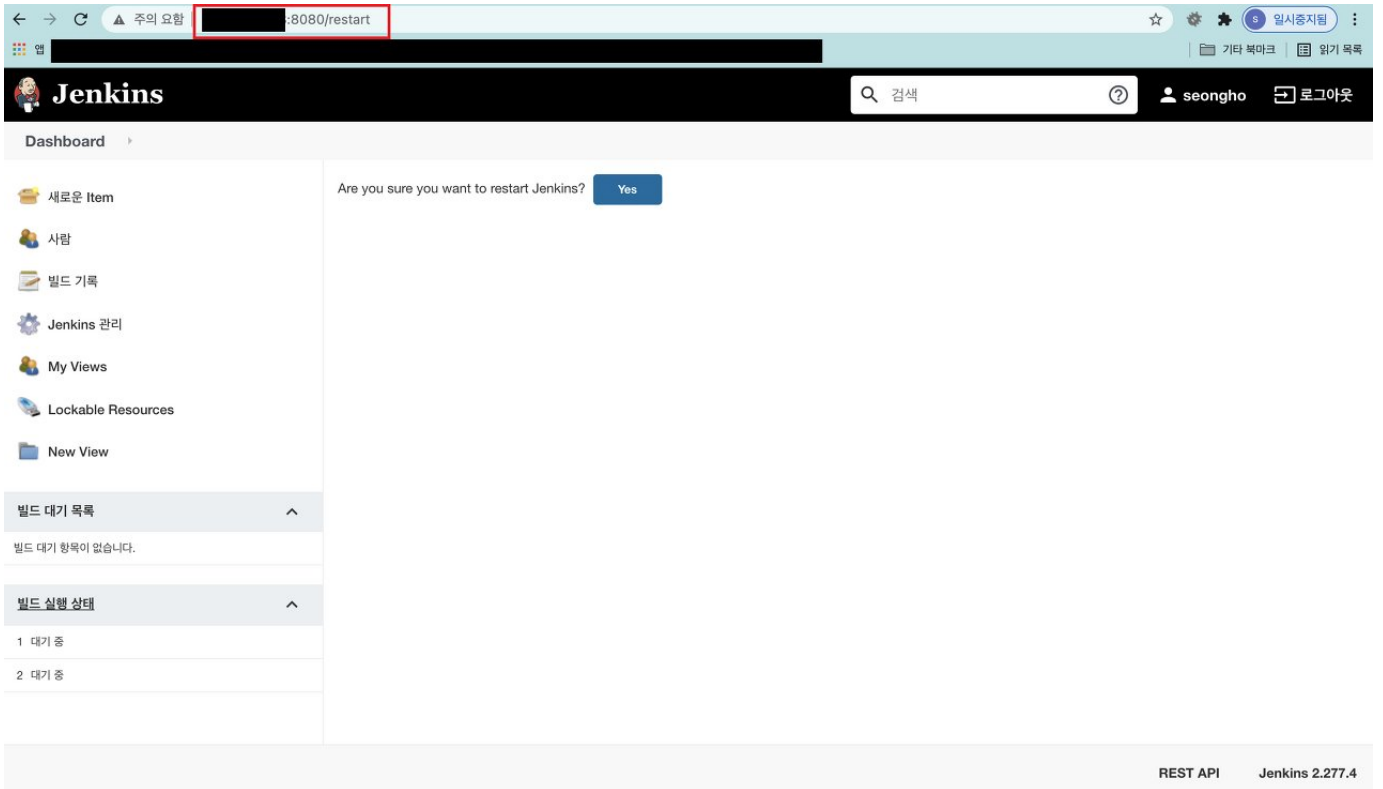
Instance Configuration

Jenkins URL:

The Jenkins URL is used to provide the root URL for absolute links to various Jenkins resources. That means this value is required for proper operation of many Jenkins features including email notifications, PR status updates, and the `BUILD_URL` environment variable provided to build steps.

The proposed default value shown is **not saved yet** and is generated from the current request, if possible. The best practice is to set this value to the URL that users are expected to use. This will avoid confusion when sharing or viewing links.

9. 모두 입력되었다면 젠킨스 서버 URL을 받게 됩니다.
젠킨스 서버 URL을 복사한 뒤, 우측 하단에 Finish 버튼을 클릭합니다.



10. 9번에서 복사한 젠킨스 URL을 도메인에 입력한 뒤, /restart를 추가로 입력하여 다시 시작합니다.
yes버튼을 클릭합니다.





Please wait while Jenkins is restarting ...

Your browser will reload automatically when Jenkins is ready.

11. 재시작중이라는 문구가 노출되고 기다려줍니다.

이로써, 젠킨스 서버 구축이 완료되었습니다.

다음편에서는 Github Repository 생성을 이어서 하겠습니다.

Ref : <https://justinadpark.tistory.com/16?category=739024>

<https://12340zszs.tistory.com/33>

<https://cheese10yun.github.io/PM2/>

블로그 내용이 도움이 되셨다면 아래 광고를 클릭해주시면 감사하겠습니다. ^^

♡ 1



구독하기



카카오토리

트위터

페이스북

'[Dev. Back-End](#) > [CI & CD](#)' 카테고리의 다른 글

[Jenkins] job scheduling 설정하는 방법 (batch) (0)	2021.07.24
[Jenkins] Global 환경변수 등록하는 방법 (0)	2021.07.23
Jenkins를 Docker Container로 구축시에 TimeZone 설정하는 방법 (0)	2021.07.21
[Jenkins] docker를 이용해서 jenkins 설치하는 방법 (ubuntu) (0)	2021.07.11
[JenKins] ⑤ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)	2021.06.04
[JenKins] ④ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)	2021.06.03

[JenKins] ③ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)

2021.06.02

[JenKins] ② 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)

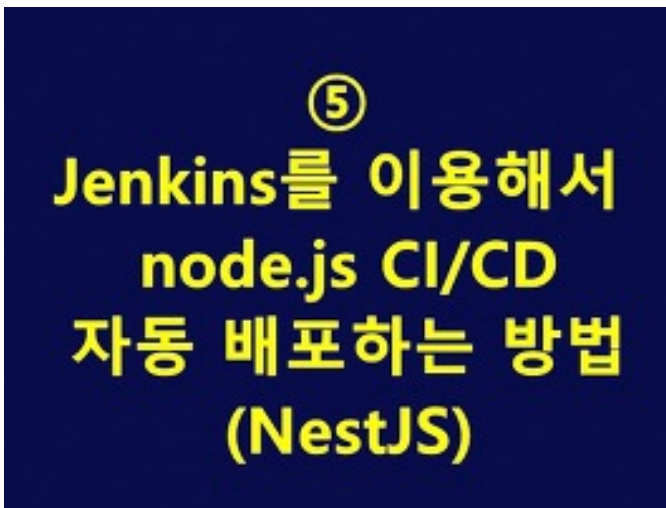
2021.06.01

[JenKins] ① 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS) (0)

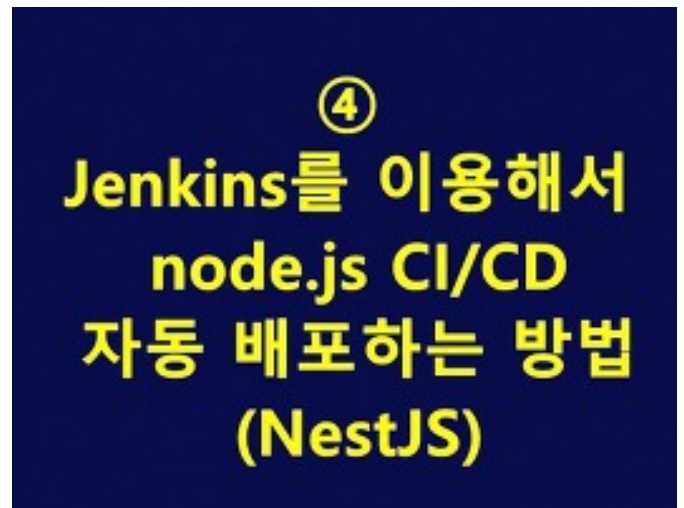
2021.05.29

태그 #AWS, #cd, #CI, #github, #jenkins, #Typescript, #배포, #배포방법, #자동화,
#젠킨스

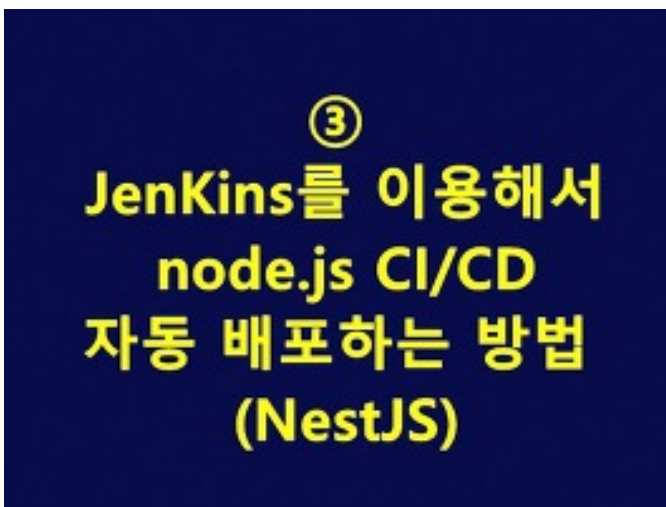
'Dev. Back-End/CI & CD' Related Articles



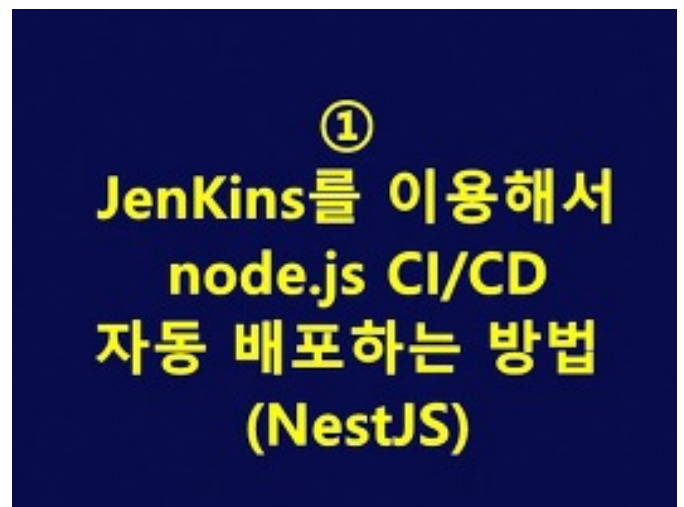
[JenKins] ⑤ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)



[JenKins] ④ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)



[JenKins] ③ 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)



[JenKins] ① 젠킨스를 이용해서 node.js CI/CD 자동 배포하는 방법 (NestJS)

이름

암호

☐

Secret

여러분의 소중한 댓글을 입력해주세요.

댓글달기

<

36/311

>