

Spring Cloud로 개발하는 マイクロ서비스 アプリケイ션



Microservices

+



Spring
Cloud

```
class Book {
    def self, title, price, author;
    self.title = title
    self.price = price
    self.author = author
}

public static void main(String[] args)
{
    var fs = require('fs');
    fs.readFile('/JONE.txt' /* 1 */,
        function (err, data) {
            console.log(data); // 3
        });
}

<@interface NextInnovationDelegate : NSObject <UIApplicationDelegate> >

<@implementation NextInnovationDelegate >
<@end>
```



목차

Part II

- **Section 0: Microservice와 Spring Cloud 소개**
- **Section 1: Service Discovery**
- **Section 2: API Gateway Service**
- **Section 3: E-commerce 애플리케이션**
- **Section 4: Users Microservice - ①**
- **Section 5: Catalogs, Orders Microservice**
- **Section 6: Users Microservice - ②**
- **Section 7: Configuration Service**
- **Section 8: Spring Cloud Bus**

Section 8.

Spring Cloud Bus

- Spring Cloud Bus
- RabbitMQ 설치
- 프로젝트 수정 – Actuator 추가
- 테스트

Previous - Changed configuration values

- 서버 재기동
- Actuator refresh
- Spring cloud bus 사용



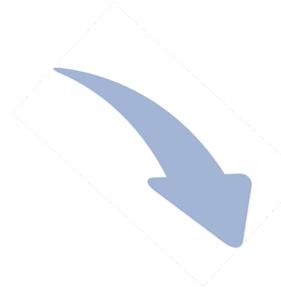
- Spring Boot Actuator
 - Application 상태, 모니터링
 - Metric 수집을 위한 Http End point 제공

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>
```



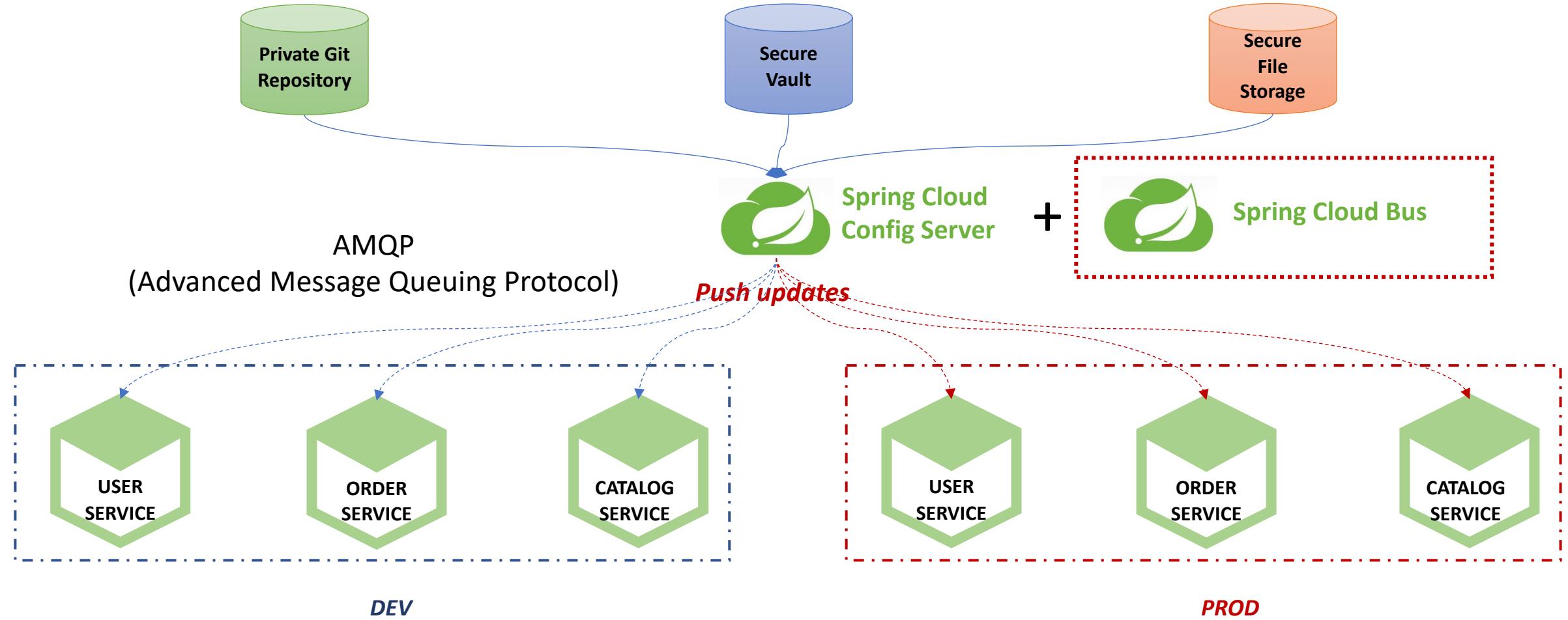
Previous - Changed configuration values

- 서버 재기동
- Actuator refresh
- Spring cloud bus 사용



- 분산 시스템의 노드를 경량 메시지 브로커와 연결
- 상태 및 구성에 대한 변경 사항을 연결된 노드에게 전달(Broadcast)

Spring Cloud Bus





Spring Cloud Bus

- AMQP (Advanced Message Queuing Protocol), 메시지 지향 미들웨어를 위한 개방형 표준 응용 계층 프로토콜
 - 메시지 지향, 큐잉, 라우팅 (P2P, Publisher-Subscriber), 신뢰성, 보안
 - Erlang, RabbitMQ에서 사용
- Kafka 프로젝트
 - Apache Software Foundation이 Scalar 언어로 개발한 오픈 소스 메시지 브로커 프로젝트
 - 분산형 스트리밍 플랫폼
 - 대용량의 데이터를 처리 가능한 메시징 시스템

Spring Cloud Bus

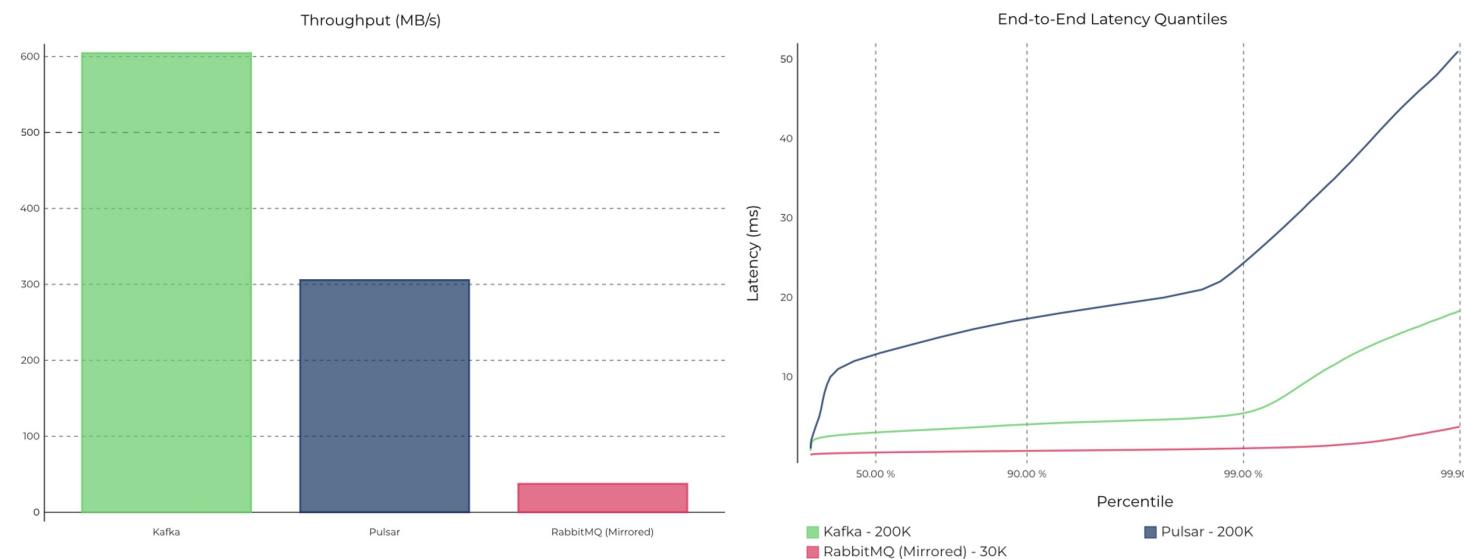


vs. kafka[®]

- RabbitMQ
 - 메시지 브로커
 - 초당 20+ 메시지를 소비자에게 전달
 - 메시지 전달 보장, 시스템 간 메시지 전달
 - 브로커, 소비자 중심
- Kafka
 - 초당 100k+ 이상의 이벤트 처리
 - Pub/Sub, Topic에 메시지 전달
 - Ack를 기다리지 않고 전달 가능
 - 생산자 중심

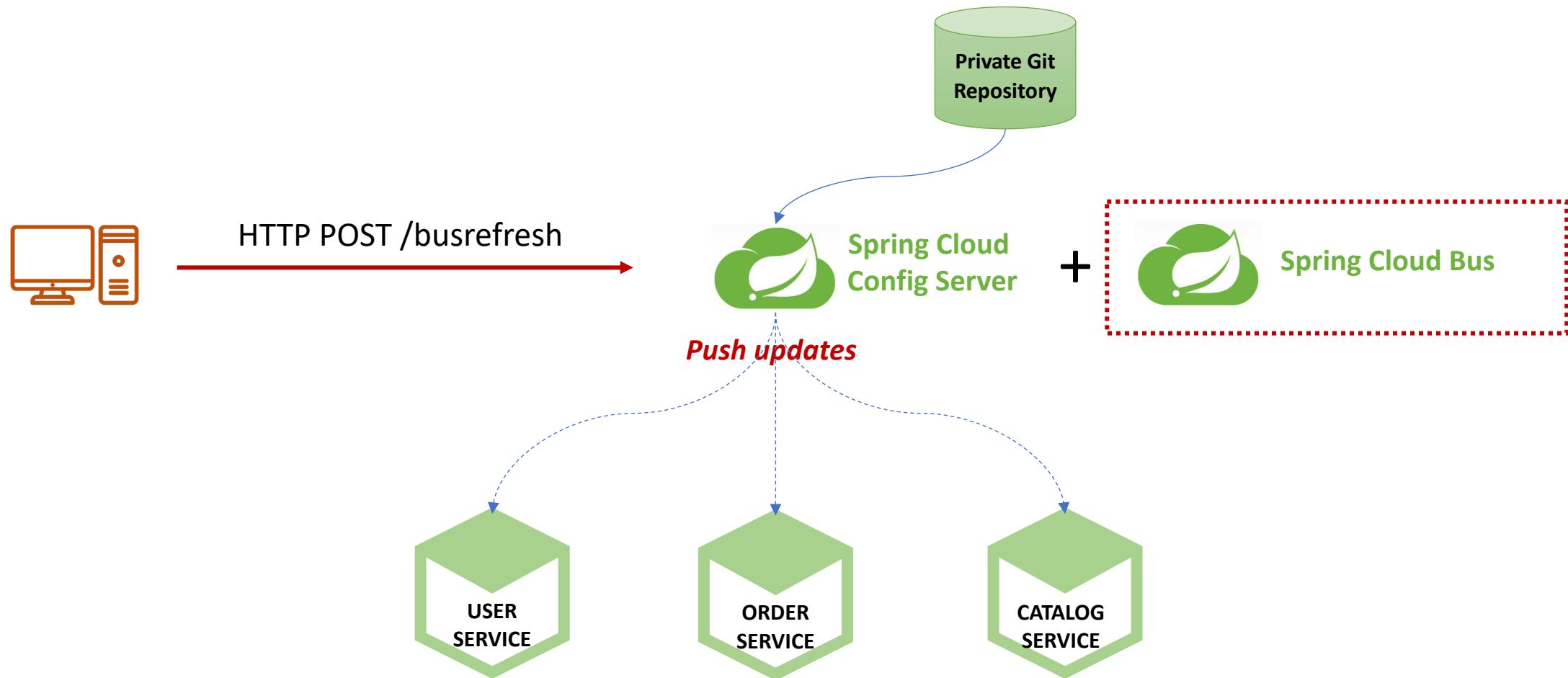
	Kafka	Pulsar	RabbitMQ (Mirrored)
Peak Throughput (MB/s)	605 MB/s	305 MB/s	38 MB/s
p99 Latency (ms)	5 ms (200 MB/s load)	25 ms (200 MB/s load)	1 ms* (reduced 30 MB/s load)

*RabbitMQ latencies degrade significantly at throughputs higher than the 30 MB/s. Furthermore, the impact of mirroring is significant at higher throughput and better latencies can be achieved by using just classic queues without mirroring.



Actuator bus-refresh Endpoint

- 분산 시스템의 노드를 경량 메시지 브로커와 연결
- 상태 및 구성에 대한 변경 사항을 연결된 노드에게 전달(Broadcast)



Rabbit MQ 설치



Features

Get Started

Support

Community

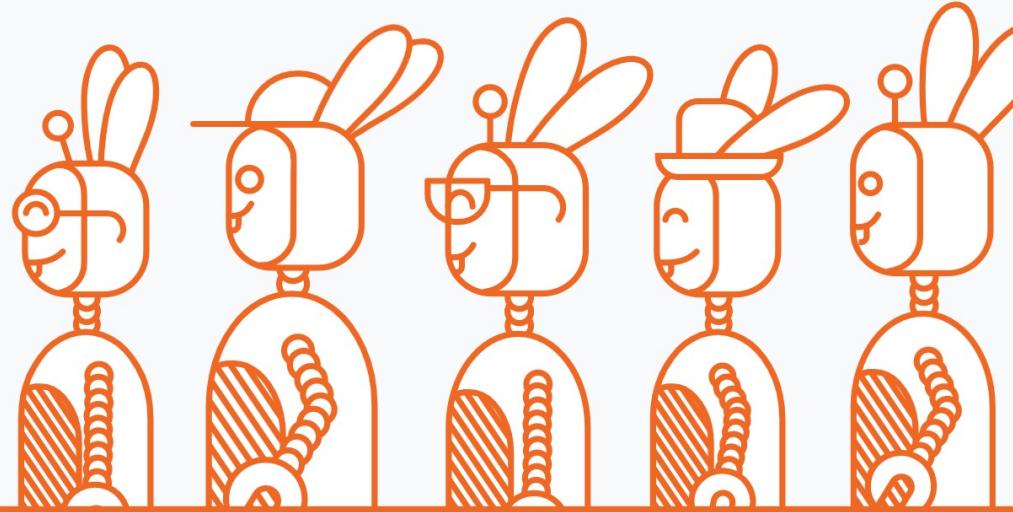
Docs

Blog

Quorum queues

A webinar on high availability
and data safety in messaging

[Learn more](#)



RabbitMQ is the most widely deployed open source message broker.

With tens of thousands of users, RabbitMQ is one of the most popular open source message brokers. From [T-Mobile](#) to [Runtastic](#), RabbitMQ is used worldwide at

Updates

1. [RabbitMQ 3.8.11](#) 22 Jan 2021
2. [RabbitMQ 3.8.10](#) 19 Jan 2021
3. [RabbitMQ 3.8.9](#) 28 Sep 2020



Rabbit MQ 설치

■ MacOS)

- \$ brew update
- \$ brew install rabbitmq

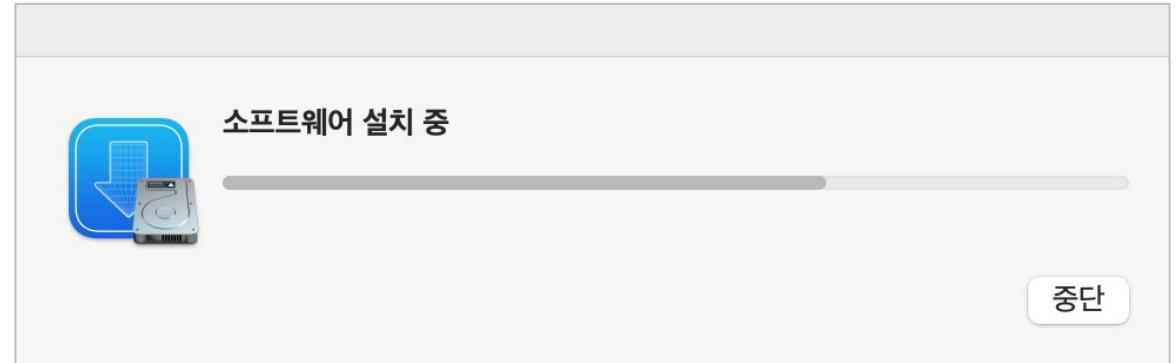
```
▶ brew install rabbitmq
==> Downloading https://homebrew.bintray.com/bottles/pkg-config-0.29.2_3.big_sur
==> Downloading from https://d29vzk4ow07wi7.cloudfront.net/0040b6ebe07f60549800b
#####
# 100.0%
==> Downloading https://homebrew.bintray.com/bottles/gdbm-1.18.1_1.big_sur.bottle
==> Downloading from https://d29vzk4ow07wi7.cloudfront.net/36b492f1b0910367dd394
#####
# 100.0%
==> Downloading https://homebrew.bintray.com/bottles/openssl%401.1-1.1.1i.big_su
==> Downloading from https://d29vzk4ow07wi7.cloudfront.net/8008537d37a7f09eedbcd
#####
# 100.0%
==> Downloading https://homebrew.bintray.com/bottles/readline-8.1.big_sur.bottle
==> Downloading from https://d29vzk4ow07wi7.cloudfront.net/2cc3a9582e3c7e21eb3c2
```

```
==> Installing dependencies for rabbitmq: python@3.9, jpeg, libpng, libtiff, wxmac and erlang
==> Installing rabbitmq dependency: python@3.9
Error: The following formula cannot be installed from bottle and must be
built from source.
  python@3.9
Install the Command Line Tools:
  xcode-select --install
```

Rabbit MQ 설치

- MacOS)

- \$ brew update
- \$ brew install rabbitmq



```
▶ xcode-select --install
xcode-select: note: install requested for command line developer tools
```

```
==> rabbitmq
Management Plugin enabled by default at http://localhost:15672

To have launchd start rabbitmq now and restart at login:
  brew services start rabbitmq
Or, if you don't want/need a background service you can just run:
  rabbitmq-server
```



Rabbit MQ 설치

- MacOS)

- \$ export PATH=\$PATH:/usr/local/sbin
- \$ rabbitmq-server

```
▶ export PATH=$PATH:/usr/local/sbin
downonlee ▶ /usr/local/sbin
▶ rabbitmq-server
Configuring logger redirection

## ##      RabbitMQ 3.8.11
## ##
##### Copyright (c) 2007-2020 VMware, Inc. or its affiliates.
##### #
##### Licensed under the MPL 2.0. Website: https://rabbitmq.com
```

Rabbit MQ 설치

The screenshot shows the RabbitMQ Management Interface running at 127.0.0.1:15672. The login screen is visible on the left, and the main Overview page is on the right.

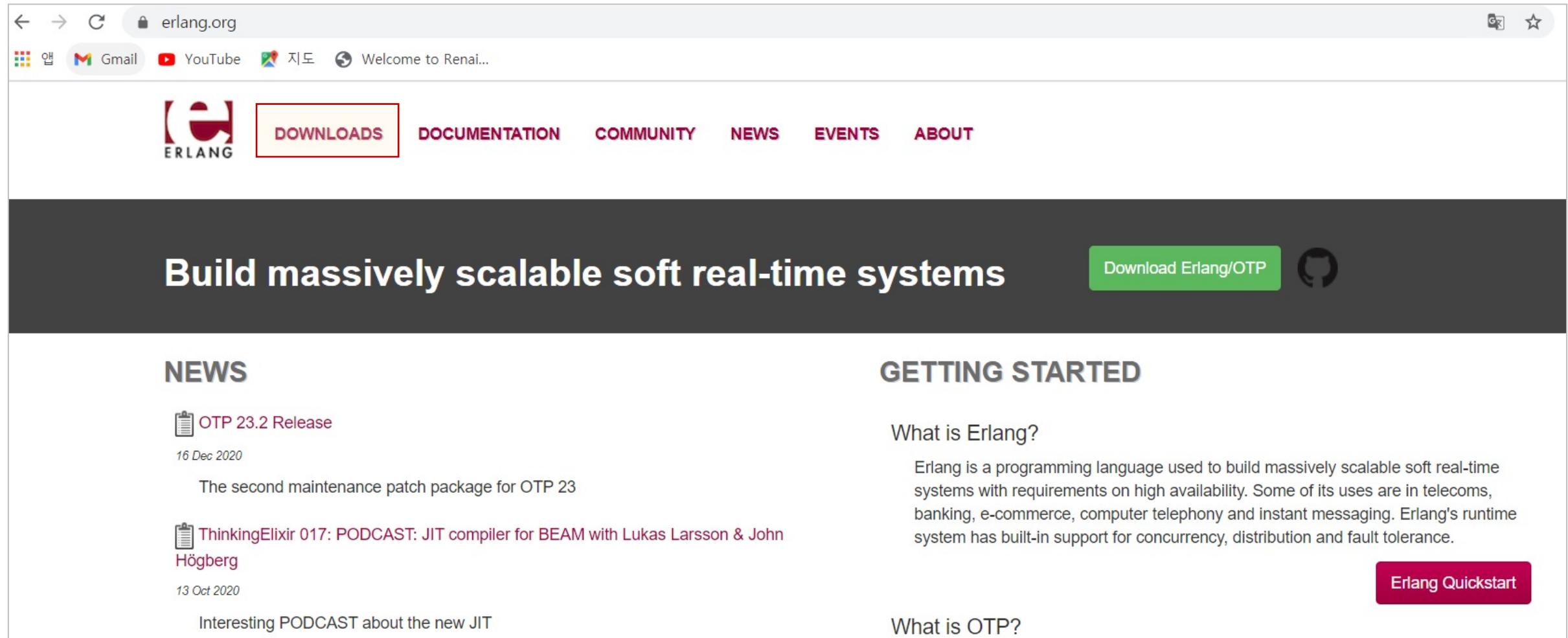
Overview Page Details:

- Header:** Refreshed 2021-02-11 18:30:03, Refresh every 5 seconds, Virtual host All, Cluster rabbit@DOWONui-MacBookPro, User guest Log out.
- Navigation:** Overview (selected), Connections, Channels, Exchanges, Queues, Admin.
- Section: Totals**
 - Queued messages: last minute ?
 - Currently idle
 - Message rates: last minute ?
 - Currently idle
 - Global counts: ?
- Metrics:** Connections: 0, Channels: 0, Exchanges: 7, Queues: 0, Consumers: 0.
- Section: Nodes**

Name	File descriptors ?	Socket descriptors ?	Erlang processes	Memory ?	Disk space	Uptime	Info	Reset stats	+/-
rabbit@localhost	121 256 available	0 141 available	497 1048576 available	86 MiB 13 GiB high watermark	671 GiB 48 MiB low watermark	3m 57s	basic disc 4 rss	This node All nodes	+/-

Rabbit MQ 설치 – Windows 10

■ Erlang 설치



The screenshot shows the Erlang.org website. At the top, there is a navigation bar with links for DOWNLOADS, DOCUMENTATION, COMMUNITY, NEWS, EVENTS, and ABOUT. The DOWNLOADS link is highlighted with a red border. Below the navigation bar, a large banner features the text "Build massively scalable soft real-time systems". To the right of the banner is a green button labeled "Download Erlang/OTP" and a GitHub icon. On the left side, there is a "NEWS" section with two entries: "OTP 23.2 Release" (published on 16 Dec 2020) and "ThinkingElixir 017: PODCAST: JIT compiler for BEAM with Lukas Larsson & John Högberg" (published on 13 Oct 2020). On the right side, there is a "GETTING STARTED" section with a "What is Erlang?" paragraph and a "What is OTP?" paragraph. A "Erlang Quickstart" button is located at the bottom right of this section.

erlang.org

Downloads Documentation Community News Events About

Build massively scalable soft real-time systems

Download Erlang/OTP

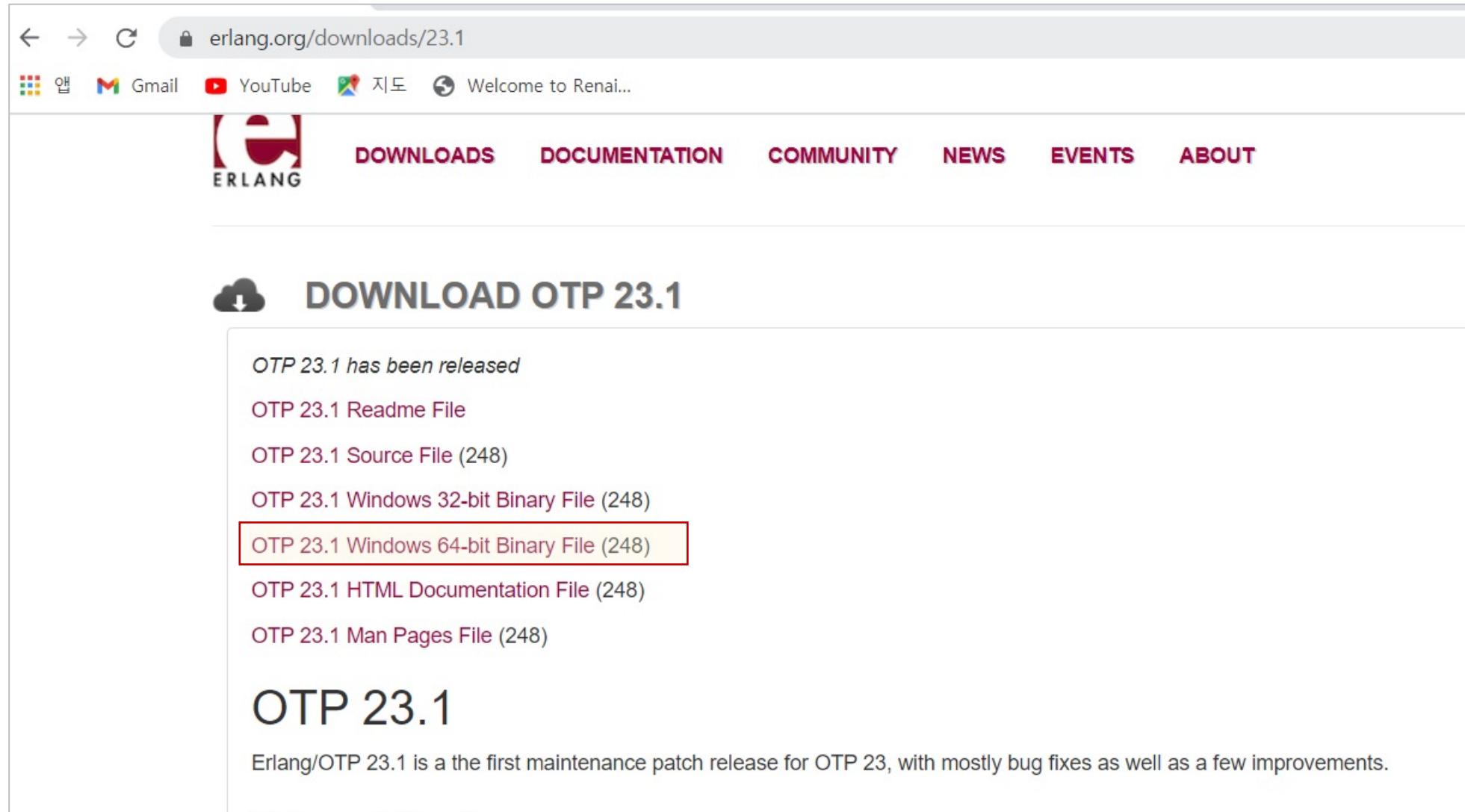
What is Erlang?

Erlang is a programming language used to build massively scalable soft real-time systems with requirements on high availability. Some of its uses are in telecoms, banking, e-commerce, computer telephony and instant messaging. Erlang's runtime system has built-in support for concurrency, distribution and fault tolerance.

What is OTP?

Erlang Quickstart

Rabbit MQ 설치 – Windows 10



← → ⌛ erlang.org/downloads/23.1

앱 Gmail YouTube 지도 Welcome to Renai...

 DOWNLOADS DOCUMENTATION COMMUNITY NEWS EVENTS ABOUT

DOWNLOAD OTP 23.1

OTP 23.1 has been released

[OTP 23.1 Readme File](#)

[OTP 23.1 Source File \(248\)](#)

[OTP 23.1 Windows 32-bit Binary File \(248\)](#)

[OTP 23.1 Windows 64-bit Binary File \(248\)](#)

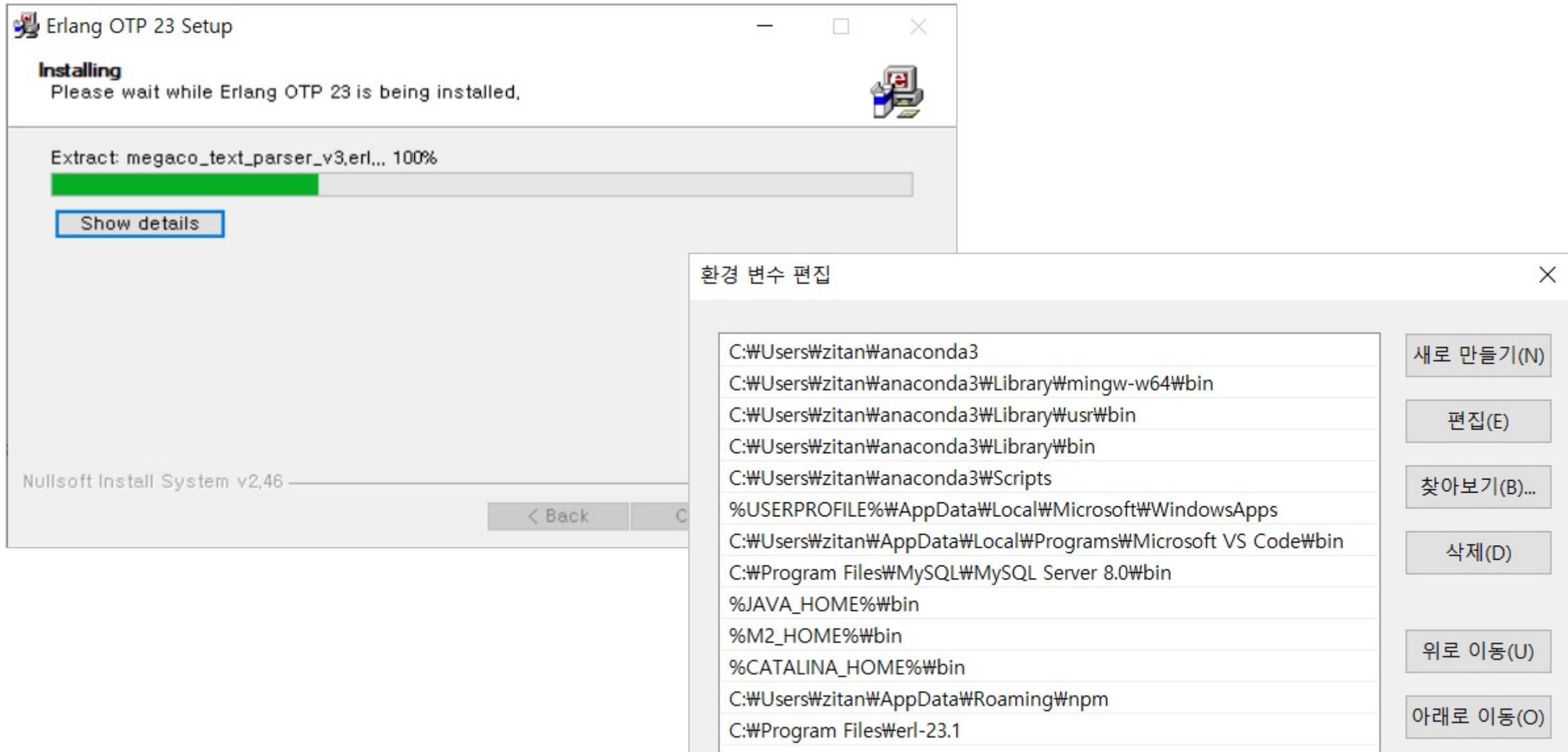
[OTP 23.1 HTML Documentation File \(248\)](#)

[OTP 23.1 Man Pages File \(248\)](#)

OTP 23.1

Erlang/OTP 23.1 is a the first maintenance patch release for OTP 23, with mostly bug fixes as well as a few improvements.

Rabbit MQ 설치 – Windows 10



Rabbit MQ 설치 – Windows 10

■ RabbitMQ 설치

The image shows two screenshots of the RabbitMQ website side-by-side. The left screenshot shows the 'Download' page ([rabbitmq.com/download.html](https://www.rabbitmq.com/download.html)) with sections for 'Downloading and Installing' and 'Installation Guides'. The right screenshot shows the 'Get Started' page ([rabbitmq.com/#getstarted](https://www.rabbitmq.com/#getstarted)) with a large 'Get Started' button, a 'Download + Installation' button, and a 'RabbitMQ Tutorials' button.

Left Screenshot (Download Page):

- Header:** rabbitmq.com/download.html
- Section 1:** RabbitMQ
- Section 2:** Downloading and Installing
- Text:** The latest [release](#) of RabbitMQ is **3.8.11**. See [change log](#) for release notes. Release series are supported.
- Text:** Experimenting with RabbitMQ on your workstation? Try the [command line interface](#).
- Text:** docker run -it --rm --name rabbitmq -p 5672:5672 -p 15672:15672 rabbitmq:3-management
- Section 3:** Open Source RabbitMQ Server
- Section 4:** Installation Guides
 - Linux, BSD, UNIX: [Debian](#), [Ubuntu](#) | [RHEL](#), [CentOS](#), [Fedora](#) | [Generic binary build](#) | [Solaris](#)
 - Windows: [Chocolatey or Installer](#) (recommended) | [Binary build](#)
 - MacOS: [Homebrew](#) | [Generic binary build](#)
 - [Erlang/OTP for RabbitMQ](#)
- Section 5:** Kubernetes
 - Open source [RabbitMQ Cluster Kubernetes Operator](#) by VMware (developed [on GitHub](#))

Right Screenshot (Get Started Page):

- Header:** rabbitmq.com/#getstarted
- Section 1:** Get Started
- Section 2:** Download + Installation
- Section 3:** RabbitMQ Tutorials
- Section 4:** Servers and clients for popular operating systems and languages
- Section 5:** Hands-on examples to get you started with RabbitMQ
- Section 6:** Installation Options
 - [Install: EC2](#)
 - [Upgrade](#)
 - [Blue-green deployment-based upgrade](#)
 - [Supported Platforms](#)
 - [Changelog](#)
 - [Erlang Versions](#)
 - [Signed Packages](#)
 - [Java Client Downloads](#)
 - [.NET Client Downloads](#)
 - [Erlang Client Downloads](#)

Rabbit MQ 설치 – Windows 10



The screenshot shows a web browser window with the URL rabbitmq.com/install-windows.html#installer. The page content includes instructions for installing Erlang and downloading the RabbitMQ installer, followed by a table of direct download links.

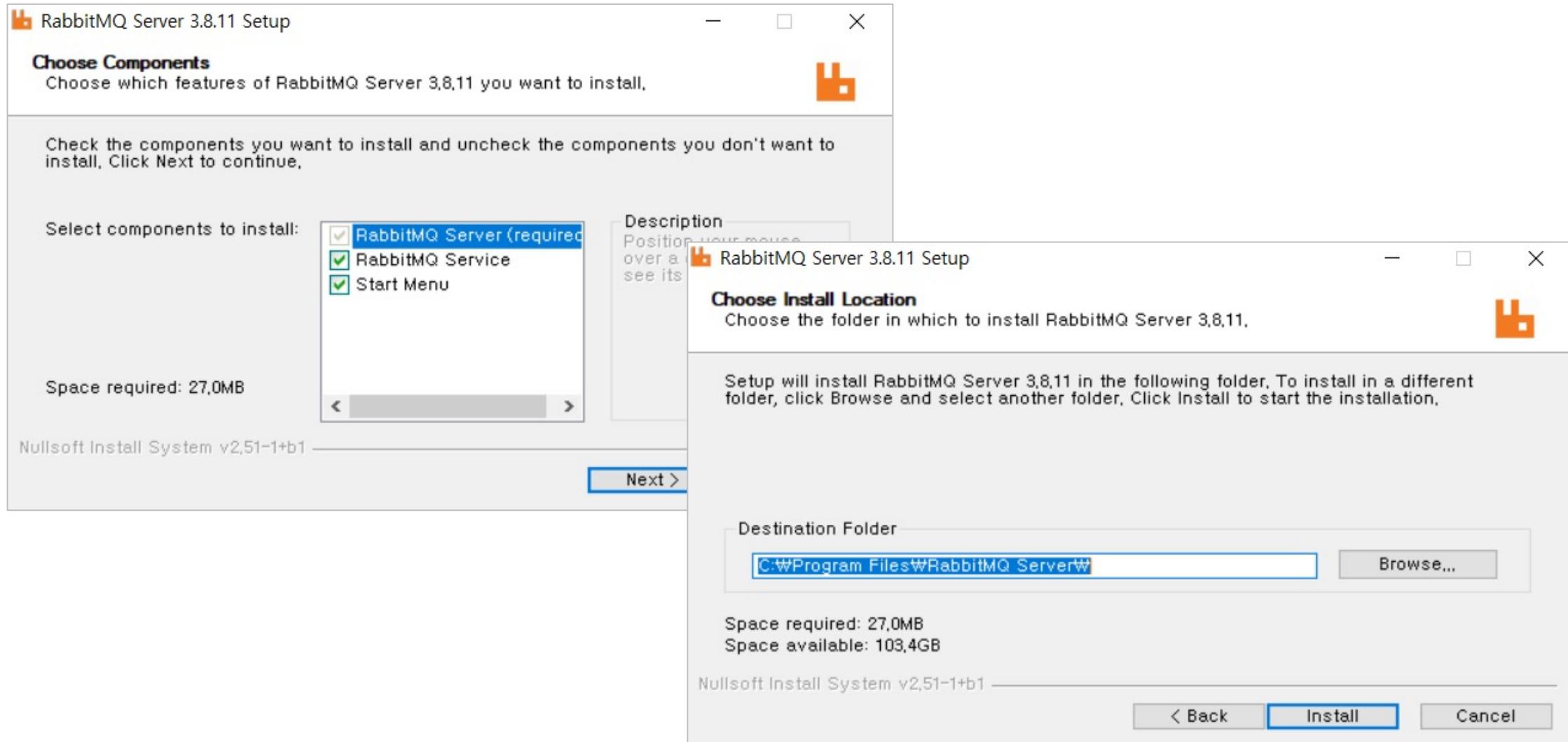
Once a supported version of Erlang is installed, download the RabbitMQ installer, `rabbitmq-server-{version}.exe` and run it. It installs RabbitMQ as a Windows service and starts it using the default configuration.

Description	Download	Signature
Installer for Windows systems (from GitHub)	rabbitmq-server-3.8.11.exe	Signature

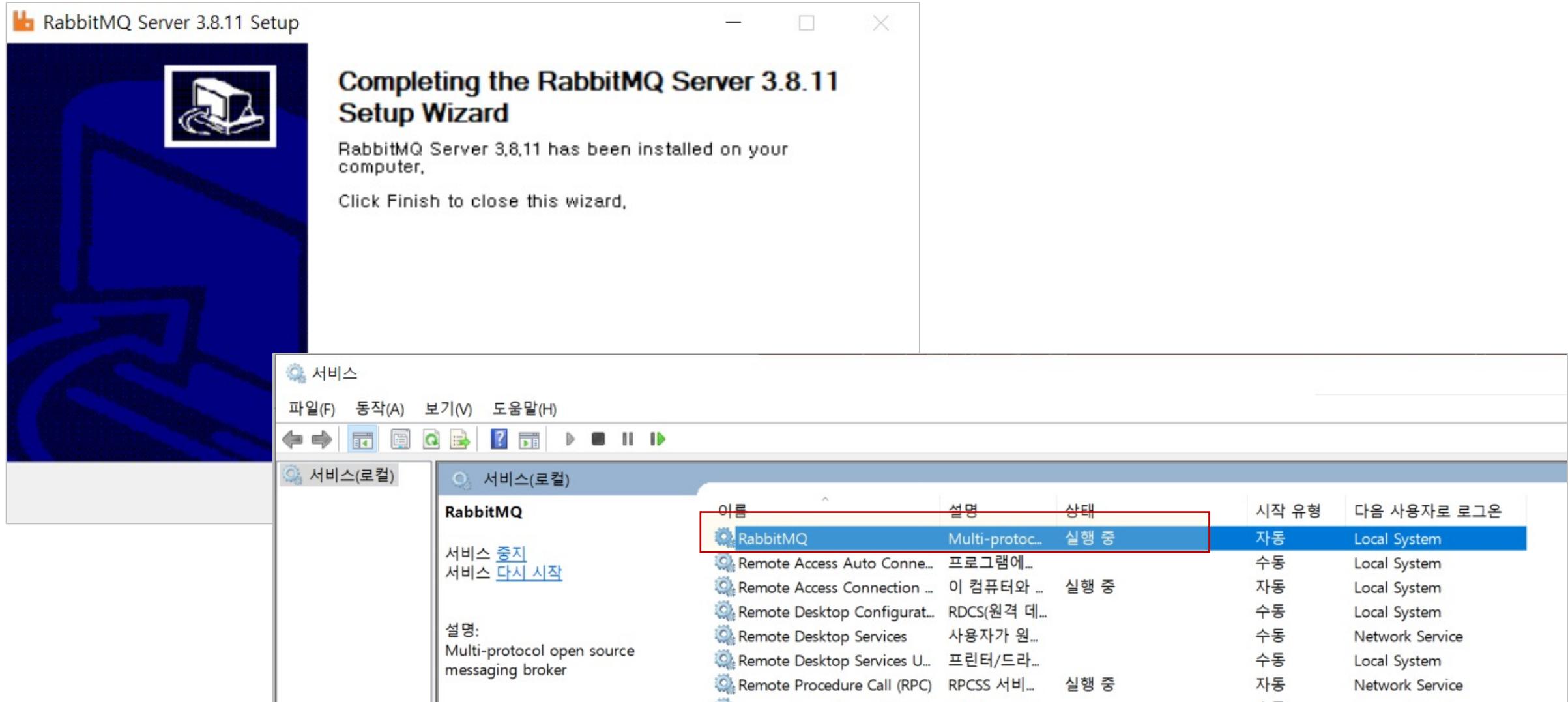
Run RabbitMQ Windows Service

Once both Erlang and RabbitMQ have been installed, a RabbitMQ node can be started as a Windows service. The RabbitMQ service starts automatically. RabbitMQ Windows service can be managed from the Start menu.

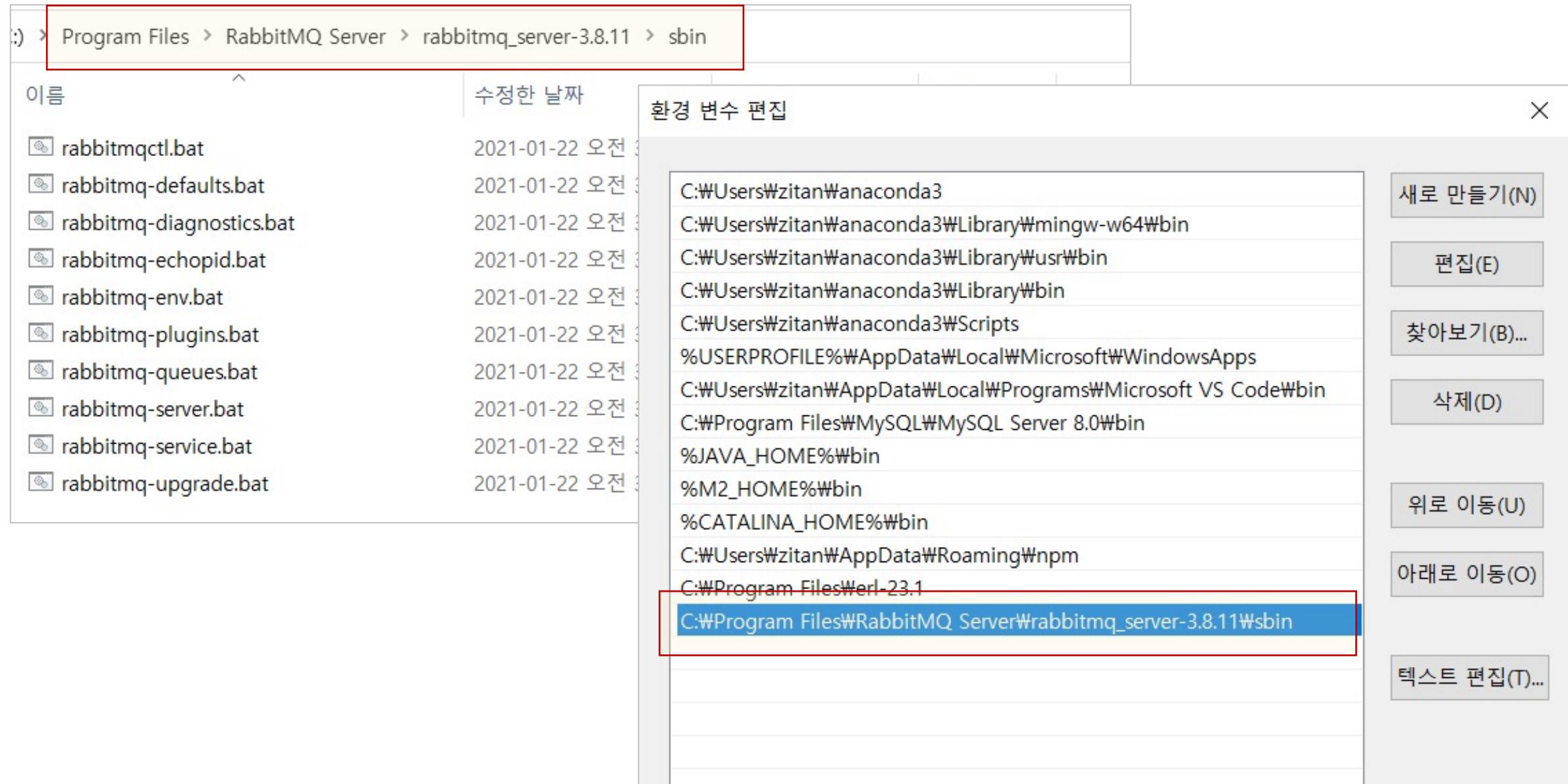
Rabbit MQ 설치 – Windows 10



Rabbit MQ 설치 – Windows 10

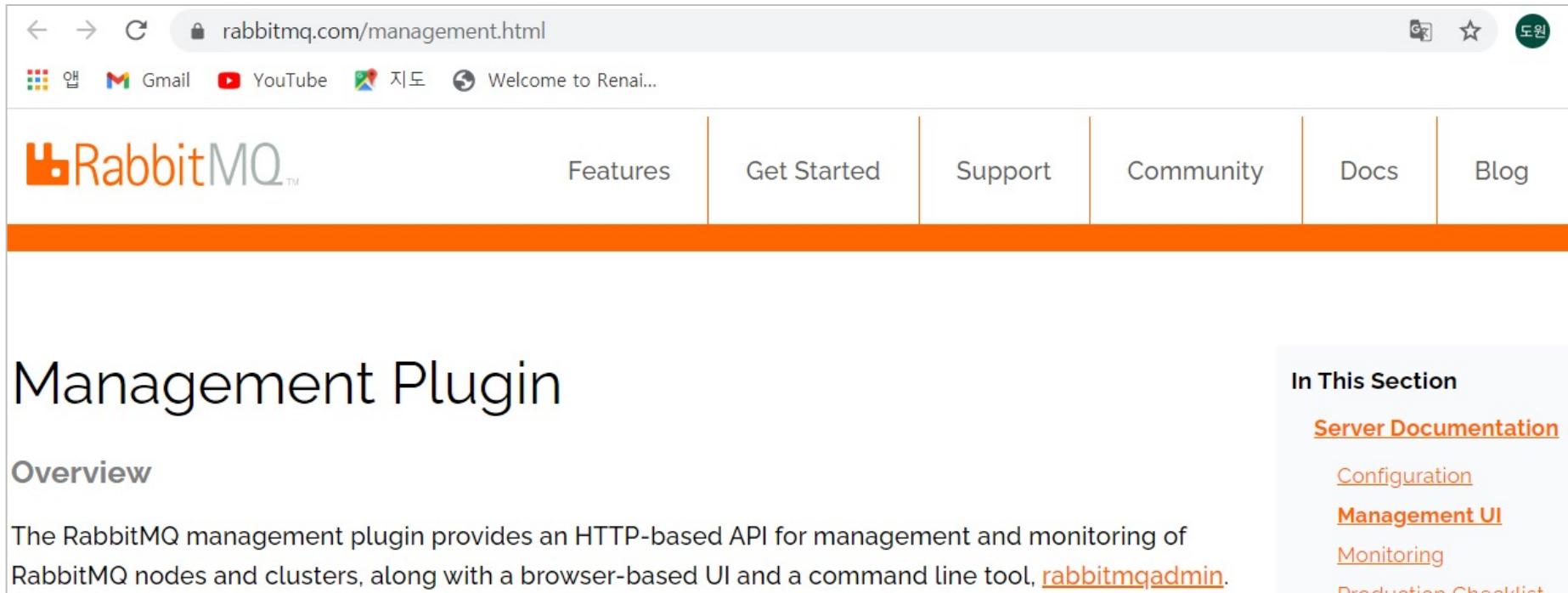


Rabbit MQ 설치 – Windows 10



Rabbit MQ 설치 – Windows 10

■ Management Plugin 설치



The screenshot shows a web browser window with the URL rabbitmq.com/management.html in the address bar. The page content is as follows:

Management Plugin

Overview

The RabbitMQ management plugin provides an HTTP-based API for management and monitoring of RabbitMQ nodes and clusters, along with a browser-based UI and a command line tool, [rabbitmqadmin](#).

It periodically collects and aggregates data about many aspects of the system to both operators in the UI and [monitoring systems](#) for long term storage, analysis and so on.

In This Section

- [Server Documentation](#)
- [Configuration](#)
- [Management UI](#)
- [Monitoring](#)
- [Production Checklist](#)

```
PS C:\Users\zitan> rabbitmq-plugins enable rabbitmq_management
Enabling plugins on node rabbit@DESKTOP-361HU6P:
rabbitmq_management
The following plugins have been configured:
rabbitmq_management
rabbitmq_management_agent
rabbitmq_web_dispatch
Applying plugin configuration to rabbit@DESKTOP-361HU6P...
Plugin configuration unchanged.
PS C:\Users\zitan>
```

Rabbit MQ 설치 – Windows 10

- <http://127.0.0.1:15672>

The screenshot shows a web browser window with the URL rabitmq.com/install-windows.html in the address bar. The page content is as follows:

Firewalls and Security Tools

Firewalls and security tools can prevent RabbitMQ Windows service and CLI tools from operating correctly. Such tools should be configured to whitelist access to [ports used by RabbitMQ](#).

Default User Access

The broker creates a user `guest` with password `guest`. Unconfigured clients will in general use these credentials. **By default, these credentials can only be used when connecting to the broker as localhost** so you will need to take action before connecting from any other machine.

See the documentation on [access control](#) for information on how to create more users and delete the `guest` user.

The screenshot shows a web browser window with the URL localhost:15672/#/ in the address bar. The page content is as follows:

RabbitMQ

Username: *

Password: *

Login

Rabbit MQ 설치 – Windows 10

- <http://127.0.0.1:15672>

The screenshot shows the RabbitMQ Management Interface running on a Windows 10 system. The browser address bar indicates the URL is `localhost:15672/#/`. The interface header includes the RabbitMQ logo, version information (RabbitMQ 3.8.11, Erlang 23.1), and a refresh button set to refresh every 5 seconds. It also shows the virtual host as 'All' and the cluster node as 'rabbit@DESKTOP-36IHU6P'. The user is logged in as 'guest'.

The main navigation menu at the top includes tabs for Overview, Connections, Channels, Exchanges, Queues, and Admin. The 'Overview' tab is currently selected.

Overview

Totals

- Queued messages last minute ?
- Currently idle
- Message rates last minute ?
- Currently idle
- Global counts ?

Metrics displayed:

- Connections: 0
- Channels: 0
- Exchanges: 8
- Queues: 0
- Consumers: 0

Nodes

Name	File descriptors ?	Socket descriptors ?	Erlang processes	Memory ?	Disk space	Uptime	Info	Reset stats	+/-
rabbit@DESKTOP-36IHU6P	0 65536 available	0 58893 available	466 1048576 available	107 MiB 3.2 GiB high watermark	101 GiB 48 MiB low watermark	9m 4s	basic disc 1 rss	This node All nodes	+/-



Dependencies 추가

- Config Server

- AMQP for Spring Cloud Bus, Actuator

```
<dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-actuator</artifactId>
</dependency>

<dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-bus-amqp</artifactId>
</dependency>
```

- Users Microservice, Gateway Service

- AMQP for Spring Cloud Bus

```
<dependency>
    <groupId>org.springframework.cloud</groupId>
    <artifactId>spring-cloud-starter-bus-amqp</artifactId>
</dependency>
```



application.yml 설정

- Config Server, Users Microservice, Gateway Service

```
spring:  
  application: <1 key>  
    rabbitmq:  
      host: 127.0.0.1  
      port: 5672  
      username: guest  
      password: guest  
  
management:  
  endpoints:  
    web:  
      exposure:  
        include: refresh,health,beans,httptrace,busrefresh
```

Spring Cloud 2020.0.0에서 bus-env → busenv, bus-refresh → busrefresh



E||스트

- 1) Start RabbitMQ Server
- 2) Start Spring Cloud Config Service
- 3) Start Eureka Discovery Service
- 4) Start Spring Cloud Gateway Service
- 5) Strat Users Microservice

GET ▾ 127.0.0.1:8000/user-service/health_check Send ▾

Params Authorization ● Headers (7) Body Pre-request Script Tests Settings

TYPE
Bearer Token

The authorization header will be automatically generated when you send the request. [Learn more about authorization](#)

Token eyJhbGciOiJIUzUxMiJ9.eyJzdWliOijmMzc0ZDjjYy1mMTI0LTQ1YWMtYTg1NS1IM

Body Cookies Headers (8) Test Results Status: 200 OK Time: 3.13 s Size: 420 B Save

Pretty Raw Preview Visualize Text ▾ ✖

1 It's Working in User Service, port(local.server.port)=49648, port(server.port)=0, with token secret=user_token_native, with token time=864000000

테스트

■ Configuration values 변경

◀ ▶ ⌂ ⓘ 127.0.0.1:8012/ecommerce/native

```
{  
    "name": "ecommerce",  
    "profiles": [  
        "native"  
    ],  
    "label": null,  
    "version": null,  
    "state": null,  
    "propertySources": [  
        {  
            "name": "file:/Users/dwonlee/Desktop/Work/native-file-repo/ecommerce.yml",  
            "source": {  
                "token.expiration_time": 864000000,  
                "token.secret": "user_token_native_changed_#1",  
                "gateway.ip": "192.168.0.7"  
            }  
        },  
    ]  
}
```

테스트

■ 이전 값 읽어 오기

The screenshot shows the Postman application interface. At the top, there is a header bar with a keyboard icon on the left, followed by the word "테스트" (Test) in large bold letters, and a search bar with the placeholder "Search" on the right.

The main area displays a request configuration:

- Method:** GET
- URL:** 127.0.0.1:8000/user-service/health_check
- Authorization:** Bearer Token

Below the URL, there are tabs for **Params**, **Authorization** (which is selected), **Headers (7)**, **Body**, **Pre-request Script**, **Tests**, and **Settings**.

In the **Authorization** tab, there is a note: "Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables." It also shows a **Token** field containing a long JWT token: eyJhbGciOiJIUzUxMjM... .

The **Body** tab is currently active, showing the response body:

```
1 It's Working in User Service, port(local.server.port)=49648, port(server.port)=0, with token secret=user_token_native, with token time=864000000
```

At the bottom, there are status indicators: Status: 200 OK, Time: 2.91 s, Size: 420 B, and a **Save** button.

테스트

- Refresh Configuration Server

- HTTP POST <http://127.0.0.1:8000/user-service/actuator/busrefresh>
 - Response code: 204 No Content

The screenshot shows the Postman application interface. At the top, there is a header bar with a keyboard icon on the left and the title "테스트" in large Korean characters. Below the header, the main content area displays a POST request to "127.0.0.1:8000/user-service/actuator/busrefresh". The "Params" tab is selected, showing a single query parameter "Key" with the value "Value". In the bottom right corner of the main area, a red box highlights the status message "Status: 204 No Content". The bottom navigation bar includes tabs for "Body", "Cookies", "Headers (6)", and "Test Results", with "Body" being the active tab. Below the navigation bar are buttons for "Pretty", "Raw", "Preview", "Visualize", "Text" (with a dropdown arrow), and a red refresh icon.

E||스트

```
: Received remote refresh request.  
: Devtools property defaults active! Set 'spring.devtools.add-properties' to 'false' to disable  
: Fetching config from server at : http://127.0.0.1:8012  
: Located environment: name=ecommerce, profiles=[dev], label=null, version=null, state=null  
: Located property source: [BootstrapPropertySource {name='bootstrapProperties-configClient'}, BootstrapProperty  
: The following profiles are active: dev  
: Started application in 0.142 seconds (JVM running for 307.815)  
: Shutting down DiscoveryClient ...  
: Unregistering ...  
: DiscoveryClient_USER-SERVICE/user-service:d447d4142a2a3bdb7571931d73ddf030 - deregister status: 200  
...  
: The response status is 200  
: Starting heartbeat executor: renew interval is: 30  
: InstanceInfoReplicator onDemand update allowed rate per min is 4  
: Discovery Client initialized at timestamp 1613044749440 with initial instances count: 2  
: Unregistering application USER-SERVICE with eureka with status DOWN  
: Saw local status change event StatusChangeEvent [timestamp=1613044749444, current=DOWN, previous=STARTING]  
: DiscoveryClient_USER-SERVICE/user-service:76bff98074d5e216d475b614814522b3: registering service...  
: Registering application USER-SERVICE with eureka with status UP  
: Saw local status change event StatusChangeEvent [timestamp=1613044749444, current=UP, previous=DOWN]  
: Keys refreshed []  
: DiscoveryClient_USER-SERVICE/user-service:76bff98074d5e216d475b614814522b3 - registration status: 204
```

```
: Fetching config from server at : http://127.0.0.1:8012  
: Located environment: name=ecommerce, profiles=[dev], label=null, version=null, state=null  
: Located property source: [BootstrapPropertySource {name='bootstrapProperties-configClient'}, BootstrapProperty  
: The following profiles are active: dev  
: Started application in 0.109 seconds (JVM running for 316.649)  
: Shutting down DiscoveryClient ...  
: Unregistering ...  
: DiscoveryClient_GATEWAY-SERVICE/192.168.0.8:gateway-service:8000 - deregister status: 200  
...  
: The response status is 200  
: Starting heartbeat executor: renew interval is: 30  
: InstanceInfoReplicator onDemand update allowed rate per min is 4  
: Discovery Client initialized at timestamp 1613044749422 with initial instances count: 2  
: Unregistering application GATEWAY-SERVICE with eureka with status DOWN  
: Saw local status change event StatusChangeEvent [timestamp=1613044749424, current=DOWN, previous=STARTING]  
: DiscoveryClient_GATEWAY-SERVICE/192.168.0.8:gateway-service:8000: registering service...  
: Registering application GATEWAY-SERVICE with eureka with status UP  
: Saw local status change event StatusChangeEvent [timestamp=1613044749425, current=UP, previous=DOWN]  
: DiscoveryClient_GATEWAY-SERVICE/192.168.0.8:gateway-service:8000 - registration status: 204
```

테스트

- /health_check 이전 값 읽어 오기 → 401 Unauthorized

The screenshot shows the Postman application interface with the following details:

- Method:** GET
- URL:** 127.0.0.1:8000/user-service/health_check
- Authorization tab (selected):** Type: Bearer Token
- Headers (7):** A warning message states: "Heads up! These parameters hold sensitive data. To keep this data secure while work recommend using variables. [Learn more about variables](#)".
- Body tab:** Token: eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJmMzc0ZDjJ'
- Status:** Status: 401 Unauthorized
- Body Content:** 1

테스트

■ 다시 로그인

POST ▼ http://127.0.0.1:8000/user-service/login

Params Authorization Headers (8) **Body** ● Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL **JSON** ▼

```
1 {
2   "email": "edowon0623@test.com",
3   "password": "test1234"
4 }
```

Body Cookies Headers (9) Test Results  Status: 200 OK

KEY	VALUE
token ⓘ	eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJmMzc0ZDjj
userId ⓘ	f374d2cc-f124-45ac-a855-e05d72ad3844

테스트

- /health_check 값 다시 읽어 오기 → 200 OK

The screenshot shows the Postman interface with the following details:

- Method:** GET
- URL:** 127.0.0.1:8000/user-service/health_check
- Authorization:** Bearer Token
- Headers:** (7)
- Body:** (Empty)
- Pre-request Script:** (None)
- Tests:** (None)
- Settings:** (None)
- Cookies:** (None)
- Notes:** A warning message: "Heads up! These parameters hold sensitive data. To keep this data secure while working in a collaborative environment, we recommend using variables. [Learn more about variables](#)"
- Token:** eyJhbGciOiJIUzUxMjM9.eyJzdWlIjMzc0ZDjjYy1mMTI0LTQ1YWMtYTg1NS1IMDVkN:...
- Status:** 200 OK
- Time:** 1970 ms
- Size:** 431 B
- Save Response:** (Link)
- Body Content:** It's Working in User Service, port(local.server.port)=49648, port(server.port)=0, with token secret=user_token_native_changed_#1, with token time=864000000

테스트

- HTTP POST <http://127.0.0.1:8000/actuator/busrefresh>
 - HTTP GET http://127.0.0.1:8000/user-service/health_check

```
try {
    subject = Jwts.parser().setSigningKey(env.getProperty("token.secret"))    subject: null
        .parseClaimsJws(jwt).getBody()
        .getSubject();
}
```

atatewayserviceApplication x

Console Endpoints | ⌂ ⌃ ⌄ ⌅ ⌆ ⌇ ⌈ ⌉ ⌊ ⌋

Variables

65, Authorization
iply\$0:43, Authori
3171167 (com.ex
rderedGatewayF
ter\$0:117, Filteri
254310 (org.spr

> `env.getProperty("token.secret") = "user_token_native_changed_#2"`

> `this = {AuthorizationHeaderFilter@11490} "[AuthorizationHeaderFilter@531ec978 configClass = AuthorizationHeaderFilter, env = StandardReactiveWebEnvironment@11493, filterIndex = 0, id = 65, name = Authorization, tokenSecret = user_token_native_changed_#2]"`

> `p jwt = "eyJhbGciOiJIUzUxMiJ9.eyJzdWliOiJmMzc0ZDJjYy1mMTI0LTQ1YWMtYTg1NS1lMDVkNzJhZDM4NDQiLCi..."`

`o1 returnValue = true`

`o1 subject = null`

> `env = {StandardReactiveWebEnvironment@11493} "StandardReactiveWebEnvironment {activeProfiles=[dev], contextPath = /, defaultContentType = application/json, defaultEncoding = UTF-8, defaultLocale = en-US, defaultPageSize = 20, defaultUriEncoding = ISO-8859-1, errorPagePath = /error, features = [ContentNegotiationFeature@11494, DefaultHandlerFeature@11495, FormHttpMessageConverter@11496, HeaderWriterFilter@11497, Jackson2JsonHttpMessageConverter@11498, MediaTypeMappingsFeature@11499, PathVariableFeature@11499, ResourceNotFoundExceptionMapper@11499, UriTemplateHttpMessageConverter@11499], featureOrder = [ContentNegotiationFeature@11494, DefaultHandlerFeature@11495, Jackson2JsonHttpMessageConverter@11498, MediaTypeMappingsFeature@11499, PathVariableFeature@11499, UriTemplateHttpMessageConverter@11499], featuresByName = {ContentNegotiationFeature@11494 = ContentNegotiationFeature@11494, DefaultHandlerFeature@11495 = DefaultHandlerFeature@11495, Jackson2JsonHttpMessageConverter@11498 = Jackson2JsonHttpMessageConverter@11498, MediaTypeMappingsFeature@11499 = MediaTypeMappingsFeature@11499, PathVariableFeature@11499 = PathVariableFeature@11499, UriTemplateHttpMessageConverter@11499 = UriTemplateHttpMessageConverter@11499}, id = 65, name = Authorization, tokenSecret = user_token_native_changed_#2}"`

Change default RabbitMQ password



RabbitMQ 3.8.11 Erlang 23.2.4

Overview Connections Channels Exchanges Queues Admin

User: guest

▼ Overview

Tags administrator

Can log in with password •

▼ Permissions

Current permissions

Virtual host	Configure regexp	Write regexp	Read regexp	
/	.*	.*	.*	<button>Clear</button>

Users

► All users

▼ Add a user

Username: admin *

Password: *

*

* (confirm)

Tags:

administrator

Set **Admin | Monitoring | Policymaker**
Management | Impersonator | None

Add user

▼ Update this user

Password: *

*

* (confirm)

Tags:

administrator

[Admin] [Monitoring] [Policymaker] [Management]

?

Update user



Actuator

- Stop RabbitMQ server

GET ▾ 127.0.0.1:8000/actuator/health

Params Authorization Headers (6) Body Pre-request Script Tests Settings

Query Params

KEY	VALUE	DESCRIPTION
Key	Value	Description

Body Cookies Headers (2) Test Results

Pretty Raw Preview Visualize JSON ▾

1 {
2 "status": "DOWN"
3 }

🌐 Status: 503 Service Unavailable T

- Start RabbitMQ server again

Pretty Raw Preview Visualize JSON ▾

1 {
2 "status": "UP"
3 }