

VI. 매니지먼트

2. packaging

목차

- 2.1 ucloud Packaging 서비스 소개
- 2.2 ucloud Packaging FAQ
- 2.3 ucloud Packaging 이용방법
- 2.4 ucloud Packaging 레시피 제작

2.1 ucloud Packaging 서비스 소개

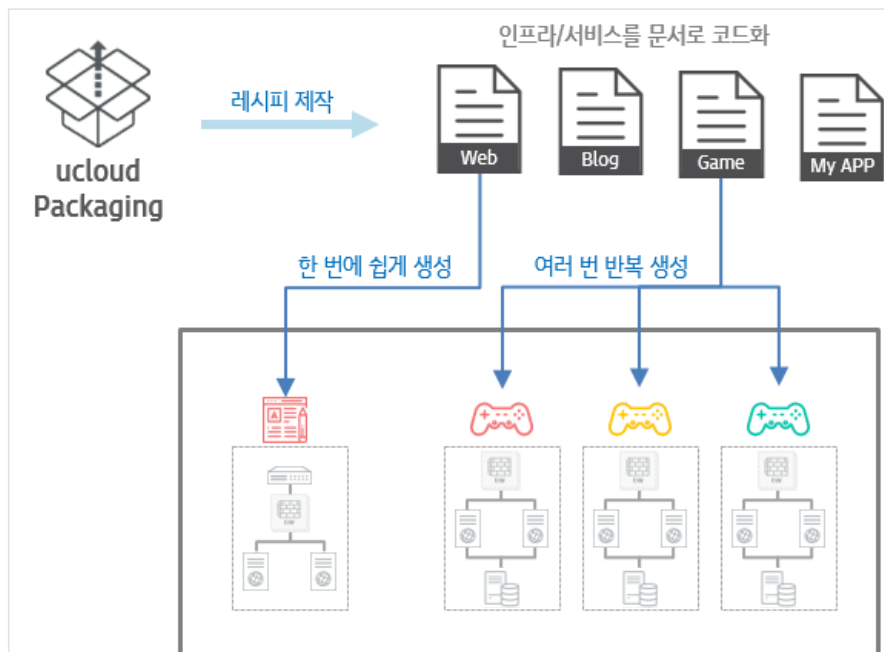
1.1.1 목적/용도

ucloud Packaging은 클라우드의 여러 자원들을 하나의 패키지로 묶어서 한 번에 배포하고 회수할 수 있습니다.

순쉬운 사용을 위해 KT에서 제공하는 완성된 서비스 형태의 패키지를 바로 사용할 수 있습니다. (예: 웹 서비스, DB 클러스터링 등)

또는 사용자가 미리 정의를 해 둔 커스텀 서비스를 한 번에 배포/회수/반복배포 할 수 있습니다.

▣ 시스템 구성도



1.1.2 구조/원리

레시피는 여러 종류의 Cloud 인프라/서비스 자원을 코드화한 JSON 문서입니다. 레시피에는 만들고 싶은 자원의 종류뿐만 아니라, 자원의 생성 순서, 연동, 결과값에 대한 처리도 함께 명시됩니다.

KT 제공 혹은 사용자가 사전에 만들어 둔 레시피를 읽어들이어서 여러 자원들을 일괄 생성하고 조합합니다.

1.1.3 유의사항/제약사항

□ Zone별 사용 가능 여부

기능	Central-A	Central-B	Seoul-M	Seoul-M2	Cloud HA	JPN	US	특수 Cloud (Enterprise, VPC 등)
ucloud Packaging	O	O	O	O	O	O	O	X

□ 공유용 레시피의 사용

공유용 레시피의 가장 큰 제공 목적은 사용자가 공유용 레시피를 참고하고 커스텀하여 개인용 레시피 제작에 활용하는 데에 있습니다.

간단하고 대표적인 레시피는 서비스 생성에 바로 사용하여도 됩니다.

하지만 다른 클라우드 자원이 업데이트/제거된 내역이 공유용 레시피에 정의된 자원 혹은 파라미터가 현행화되지 않아 정상 생성이 되지 않을 수 있습니다.

공유용 레시피가 패키지 생성에 그대로 사용되지 못함을 장애로 분류하지 않습니다.

생성한지 오래되어 정상 동작하지 않는 공유용 레시피가 있을 경우 리포트 해 주시기 바랍니다.

□ 개인용 레시피의 작성과 사용

패키지 생성을 위한 레시피를 사용자가 직접 작성하여 사용할 경우, 레시피의 작성이 잘못되었거나 패키지 생성 시 파라미터의 입력이 잘못되어 패키지 생성이 실패할 수 있습니다.

사용자가 직접 생성한 개인용 레시피 및 입력할 파라미터가 정상적인지 여부는 사용자가 직접 테스트 및 판단해야 합니다. (패키지 생성 시 롤백 기능을 OFF한 후 이벤트 로그 확인 필요)

Packaing 서비스 내에서 기본적으로 레시피의 json 문법 검사는 수행합니다만 (레시피 등록 과정 혹은 OpenAPI-validateRecipe)

실제 내부 설정 값이나 파라미터 등 전체적인 정합성에 대한 모든 검사를 수행하지는 않습니다.

또한 레시피는 아래와 같은 용량 제한이 있습니다.

- 포탈 UI를 통한 업로드: 20KB 이하

- OpenAPI를 통한 업로드: 8KB 이하

보다 큰 레시피 파일의 업로드를 시도하는 경우 에러가 발생합니다.

▣ 패키지의 삭제 (회수)

패키지를 삭제를 시작하면, 과정 중에 패키지 내 포함되어 있는 모든 자원을 삭제해야 패키지도 정상 삭제 됩니다.

패키지 내 자원이 특정 이유로 삭제가 불가능하면 패키지도 삭제가 되지 않습니다. (예: ucloud DB의 이중화 상태 불일치)

이를 해소하기 위해 패키지 강제 삭제 기능을 제공하고 있습니다.

패키지 내 자원이 삭제 불가능하더라도 Skip 하고 계속 삭제를 진행합니다.

이 과정에서 패키지에 소속된 클라우드 자원 일부가 삭제되지 않을 수 있습니다.

패키지를 삭제 후에는 소속된 클라우드 자원이 모두 삭제되었는지 추가 확인이 필요합니다.

삭제되지 않고 남은 자원에 대한 지속되는 과금은 사용자의 책임이 있습니다.

▣ 레시피를 업로드하거나 검사를 하면 올바르지 않다고 오류가 나옵니다.

올바르지 않은 경우는 아래 중 하나입니다.

1. 실제 json 문법이 올바르지 않은 부분이 있어 그렇습니다. 특히 괄호 내 쉼표 쓰임에 주의해야 합니다.

2. 문법이 모두 옳음에도 안되는 경우는 레시피 body가 길이 제한을 넘은 경우 입니다.

- 포탈 UI로 업로드 및 검사할 경우 20KB 이하

- OpenAPI로 업로드 및 검사할 경우 7KB 이하

▣ 패키지 삭제가 실패합니다.

실패한 패키지를 누른 후 아래 상세 내역 중 '이벤트'를 보시면 삭제 실패한 원인을 알 수 있습니다.

1. 삭제 실패한 원인을 제거한 후 다시 삭제를 진행합니다.

2. 혹은 패키지에 포함된 자원을 별도로 삭제(예정)인 경우, 강제 삭제를 통해 패키지를 삭제합니다. (강제 삭제의 경우, 패키지 내 자원이 완벽히 삭제되지 않고 남을 수 있습니다.)

2.3 ucloud Packaging 이용방법

2.3.1 Packaging 정의

ucloud Packaging에서 사용되는 용어 및 개념에 대해 소개 합니다.

▣ 패키지

Packaging 서비스를 통해 실제 클라우드에 일괄 생성된 여러 자원들의 집합입니다.

하나의 어플리케이션 서비스 단위라고 이해해도 됩니다.

▣ 레시피

원하는 자원들을 조합하여 패키지를 만들기 위해서, 사전에 코드 형태로 정의한 문서를 레시피라 합니다.

레시피에는 클라우드 자원의 정보, 소프트웨어 설치, 자원들간의 관계 정의 등 패키지 배포에 필요한 여러 정보들이 들어 있습니다.

또한 배포 시점마다 변경해야 할 파라미터들에 대한 내용도 포함되어 있습니다.

▣ 배포

레시피를 이용하여 실제 클라우드에 패키지를 만드는 작업입니다.

▣ 레시피 파라미터

하나의 레시피로 여러 번 반복하여 여러 패키지로 배포할 때, 조금씩 변경할 부분이 있을 수 있습니다. 가령 Server의 사양 변경이나, 각각 다른 Zone에 배포하거나, 디스크의 용량을 더 크게 만들 상황이 생길 수 있습니다.

배포할 때마다 이런 변경 사항들을 파라미터로 입력 받아 조금씩 다른 패키지를 만들 수 있습니다.

▣ 자원

패키지에 포함되는 구성 요소들입니다.

쉽게 생각하면, 하나의 패키지에 포함된 Server, Disk, 로드밸런서, DB 등의 자원이 포함됩니다.

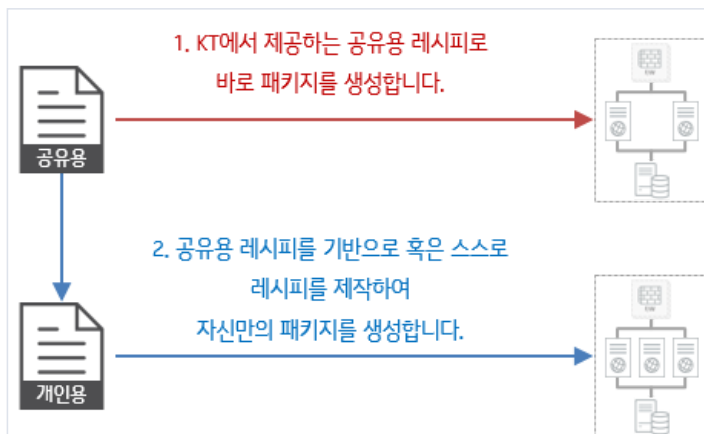
이 뿐만 아니라 네트워크 작업(Ex.포트포워딩), 자원간의 결합(Ex.Disk Attach), 규칙 설정(Ex.WAF에 웹 서버 추가) 등 모든 작업도 자원에 포함됩니다.

▣ 자원 특성

레시피 안에 자원에 대해 명시할 때 다음의 3가지의 특성을 기술할 수 있습니다.

- Type: 자원의 종류. VirtualMachine, LoadBalancer 등
- Properties : 이 자원을 클라우드에 배포하기 위해 필요한 속성 정보. template ID, zone ID, size 등
- DependsOn : 자원 배포 순서를 명시

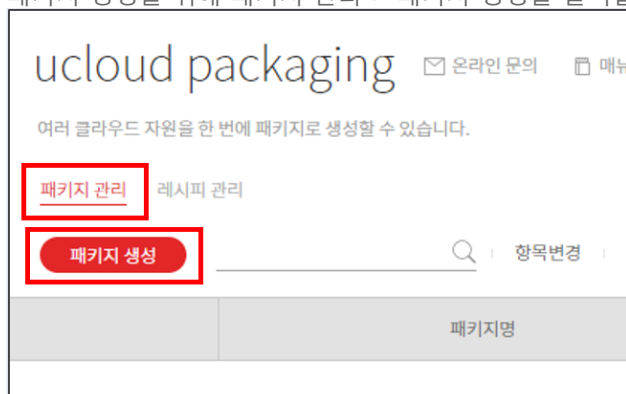
자세한 내용은 레시피 제작을 참고 바랍니다.



2.3.2 패키지 관리

▣ 패키지 생성

패키지 생성을 위해 패키지 관리 > 패키지 생성을 클릭합니다.



패키지 생성 설정을 합니다.

패키지 생성
온라인 문의
매뉴얼
ucloud

여러 클라우드 자원을 한 번에 패키지로 생성할 수 있습니다.

1. 레시피 선택 > 2. 파라미터 설정 > 3. 생성 내역 확인

1단계 : 레시피 선택

*표시는 필수 입력사항입니다.

1

* 패키지명

test

* 최대255byte 영문과숫자-,_만가능

2

* Zone

일반 Zone

*일반 Zone KOR-Central A, Central B, Seoul M, HA, JPN, US-West

3

* 레시피 선택

☐ 개인용 레시피 레시피를 선택하세요
☒ 공유용 레시피 WEB_httpd_1VM
☐ 레시피 직접 사용

4

생성 제한시간(분)

없음

5

생성 실패시 롤백 여부

☒ 예
☐ 아니오

6

완료 알림

☒ 예
☐ 아니오

토픽 URN

urn:ucloud:umsg:-2142651461:sty

토픽 생성

구독

프로토콜: SMS
수신처 010 - 0000 - 0000

취소
다음

(1) 생성할 패키지명을 입력합니다.

(2) 자원을 생성할 Zone을 선택합니다. 실제 Zone 선택은 레시피 내 정해져 있거나 아래 파라미터 입력에서 받을 수 있습니다.

- 일반 Zone: KOR Seoul-M, Central-A/B, Cloud-HA, JPN, US-West
- KOR-Seoul M2 Zone: KOR-Seoul M2

* 특수 Cloud, 특수 Zone은 지원하지 않습니다. (예: Enterprise, G-Cloud, VPC 등)

(3) 사용할 레시피를 선택합니다.

- 개인용 레시피 : 아래 레시피 관리에서 사용자가 만들어 등록한 레시피를 사용합니다.
- 공유용 레시피 : KT에서 미리 생성하여 제공하는 레시피를 바로 사용합니다.
- 레시피 직접 사용 : 레시피 저장 없이 바로 레시피를 입력하여 사용할 수 있습니다.

* 주의 : 레시피 직접 사용의 경우, 사용한 레시피가 저장되지 않습니다.

(4) 생성 제한 시간을 설정합니다.

패키지 생성 완료가 제한 시간까지 되지 않으면 생성 실패(혹은 설정에 따라 롤백)가 됩니다.

(5) 생성 실패 시 롤백 여부를 설정 합니다.

패키지 생성 중 특정 자원 생성이 잘못되거나 파라미터를 잘못 입력하는 등의 사유로 생성이 실패했을 경우, 롤백(rollback) 할지 말지를 결정합니다.

- 롤백을 설정할 경우, 패키지 생성이 실패 했을 때, 그때까지 만든 자원을 정리하고 패키지까지 모두 자동으로 삭제합니다.

- 롤백을 설정하지 않을 경우, 패키지가 생성 실패 상태로 남게 됩니다. 대신 왜 실패했는지 원인을 상세 > 이벤트 창에서 확인하여 조치가 가능합니다.

(6) 완료 알림을 받을 수 있습니다.

완료 알림 기능은 ucloud Messaging을 이용하여 발송합니다. ucloud Messaging에 대한 내용은 다음을 참고하십시오.

토픽이 없는 경우 '토픽 생성'을 클릭하여 토픽을 먼저 생성합니다.

완료 알림을 받을 프로토콜을 HTTP(S), SMS, E-mail 선택하여 설정합니다.

패키지 생성에 필요한 파라미터를 지정합니다.

어떤 파라미터를 어떻게 받을것인지는 이전 단계에서 지정한 레시피에서 이미 정해져 있습니다.

패키지 생성

온라인 문의 매뉴얼

ucloud server > ucloud

여러 클라우드 자원을 한 번에 패키지로 생성할 수 있습니다.

1. 레시피 선택 > 2. 파라미터 설정 > 3. 생성 내역 확인

2단계 : 파라미터 지정

표시는 필수 입력사항입니다.

1 패키지설명

매뉴얼 작성용 테스트 패키지입니다.

2 레시피 설명

1VM Single 웹 서버 생성 레시피

3 파라미터 설정

4 주요 파라미터

webPortNumber80[Server] 서버를 공인IP로 연결할 Web port를 입력하세요

sshPortNumber22[Server] 서버를 공인IP로 연결할 SSH port를 입력하세요

downloadURLhttp://www.naver.com[Server] 다운로드 할 웹 소스 경로를 입력하세요

ProductCodestd_cent 7.0 64bit en_1x1[Server] Product Code를 입력하세요 (CentOS)

Zoneld95e2f517-d64a-4866-8585-5177c256f7c7[Server] Zone ID를 입력하세요

취소

이전

다음

(1) 패키지 설명을 입력합니다.

패키지 별 구분을 위해 설명을 입력할 수 있습니다.

(2) 레시피 설명입니다.

현재 선택한 레시피에 대한 설명을 볼 수 있습니다. 입력이나 수정할 수 없습니다.

(3) 파라미터를 설정합니다.

레시피에서 미리 정해둔 파라미터들에 대해 실제 사용할 파라미터를 입력할 수 있습니다.

가령, ProductCode의 std_cent 7.0 64bit en_1x1 파라미터를 std_cent 7.0 64bit en_2x2 로 변경하여 더 높은 사양의 Server를 생성할 수 있습니다.

(4) 파라미터 입력에 필요한 주요 파라미터가 팝업으로 표시되어 입력 시 참고할 수 있습니다.

주요 파라미터			
Zone	Product Code	IP	CIP
Name		ID	
KOR-Central A		eceb5d65-6571-4696-875f-5a17949f3317	
KOR-Central B		9845bd17-d438-4bde-816d-1b12f37d5080	
KOR-Seoul M		95e2f517-d64a-4866-8585-5177c256f7c7	
KOR-Seoul M2		d7d0177e-6cda-404a-a46f-a5b356d2874e	
KOR-HA		dfd6f03d-dae5-458e-a2ea-cb6a55d0d994	
JPN		3e8ce14a-09f1-416c-83b3-df95af9d6308	
US-West		b7eb18c8-876d-4dc6-9215-3bd455bb05be	

설정 내역을 확인 후 생성을 클릭합니다.

패키지 생성

온라인 문의

매뉴얼

여러 클라우드 자원을 한 번에 패키지로 생성할 수 있습니다.

1. 레시피 선택 > 2. 파라미터 설정 > 3. 생성 내역 확인

3단계 : 생성 내역 확인

패키지 생성 정보

패키지명

test

패키지설명

매뉴얼 작성용 테스트 패키지입니다.

레시피 설명

1VM Single 웹 서버 생성 레시피

Zone

일반 Zone

파라미터 설정

webPortNumber

80

sshPortNumber

22

downloadURL

http://www.naver.com

ProductCode

std_cent 7.0 64bit en_1x1

ZoneId

95e2f517-d64a-4866-8585-5177c256f7c7

위의 내용과 같이 패키지를 신청하시겠습니까?

취소

이전

생성

생성 과정은 대상 패키지 클릭 > 이벤트 에서 진행 상황 확인이 가능합니다.

필요 시 진행 상황 업데이트를 위해 우측의 '새로 고침' 버튼을 활용 합니다.

✓ 선택된 패키지: test

설명 결과값 리소스 이벤트 레시피 파라미터

새로고침

시간	종류	이름	ID	상태	이유
Thu Sep 27 14:19:25	UPAC::VirtualMachine	httpdVM	4d2c440e-b6cf-4790-a605-6d240e02d688	CREATE_IN_PROGRESS	Updating physical resource id
Thu Sep 27 14:18:59	UPAC::VirtualMachine	httpdVM	httpdVM-test	CREATE_IN_PROGRESS	
Thu Sep 27 14:18:59	UPAC::IpAddress	httpdIP	115a8981-0753-4992-9422-8806983ff258	CREATE_COMPLETE	
Thu Sep 27 14:18:48	UPAC::IpAddress	httpdIP	115a8981-0753-4992-9422-8806983ff258	CREATE_IN_PROGRESS	Updating physical resource id
Thu Sep 27 14:18:47	UPAC::IpAddress	httpdIP	httpdIP-test	CREATE_IN_PROGRESS	

패키지 상세

설명

패키지에 대한 일반 설명입니다.

✓ 선택된 패키지: test

설명 결과값 리소스 이벤트 레시피 파라미터

· 패키지명

test

· 패키지 ID

urn:ucloud:upac:-2142651461:test

· 상태

● 생성완료

· 생성일시

Thu Sep 27 14:18

· 설명

매뉴얼 작성용 테스트 패키지입니다.

결과값

레시피에서 output 영역에서 정해둔 output 결과값을 확인 가능합니다.

예를 들어, 패키지를 만든 후 패키지에 포함된 VM의 비밀번호나, 생성한 IP 등 패키지 생성후 반환이 필요한 값들을 받아볼 수 있습니다.

✓ 선택된 패키지: test

설명 결과값 리소스 이벤트 레시피 파라미터

새로고침

키	값	설명
publicIP	211.253.26.142	생성된 VM으로 접근 가능한 공인IP
generatedVMpw	qA5zrcwyuxmf	생성된 VM의 SSH 비밀번호

리소스

패키지 내에서 생성된 자원(리소스) 목록을 볼 수 있습니다.

일반적으로 생각할 수 있는 자원인 Server, NAS, DB, 로드밸런서, IP 등의 실존 자원 뿐만 아니라, IP 생성, 포트포워딩 내역, Volume Attach 등의 작업도 모두 자원으로 포함됩니다.

리소스 타입에 대한 내용은 다음을 참고하시기 바랍니다.

선택한 패키지를 생성할 때 입력한 파라미터 내역을 확인할 수 있습니다.

✓ 선택된 패키지: test			
설명	결과값	리소스	이벤트 레시피 파라미터
이름	값	설명	비고
webPortNumber	80		
sshPortNumber	22		
downloadURL	http://www.naver.com		
ProductCode	std_cent 7.0 64bit en_1x1		
ZoneId	95e2f517-d64a-4866-8585-5177c256f7c7		

□ 패키지 삭제

패키지 리스트에서 삭제할 패키지를 선택 후 Action > 삭제 를 클릭합니다.

The screenshot shows the 'ucloud packaging' web interface. At the top, there's a navigation bar with 'ucloud server / ucloud packaging / 패키지 관리' and a search bar. Below the navigation bar, there's a table of packages. The table has columns: '패키지명', '생성일시', '상태', '설명', and 'Action'. One package is listed: 'test' with creation time 'Tue Oct 30 16:51' and status '생성완료'. A modal dialog titled '패키지 삭제' is open in the center, asking '해당 패키지를 삭제 하시겠습니까?' (Do you want to delete this package?). It includes a checkbox for '강제 삭제' (Force delete) and a warning: '강제 삭제 시 리소스가 삭제되지 않을 수 있습니다.' (Resources may not be deleted when forcing deletion). There are '취소' (Cancel) and '삭제' (Delete) buttons at the bottom of the dialog.

패키지 삭제는 하나의 요청입니다.

패키지 삭제는 패키지에 포함된 자원을 순차적으로 삭제합니다.

삭제 도중 패키지에 포함된 자원을 삭제할 수 없는 경우 (예: 다른 곳에서 이미 자원 삭제, DB가 삭제할 수 있는 상태가 아님 등)

그 삭제 요청은 실패로 처리되며, 실패 상태로 남게 됩니다.

○ 강제 삭제

패키지에 포함된 자원을 삭제할 수 없어 패키지를 삭제할 수 없는 경우, 강제 삭제를 수행하여 패키지를 삭제할 수 있습니다.

강제 삭제 옵션을 사용하면, 패키지에 포함된 자원을 순차적으로 삭제 시도는 하지만 실패하더라도 다음 과정으로 넘어갑니다.

즉, **패키지는 무조건 삭제가 되지만, 패키지에 소속된 자원이 남아 있을 수 있습니다.**

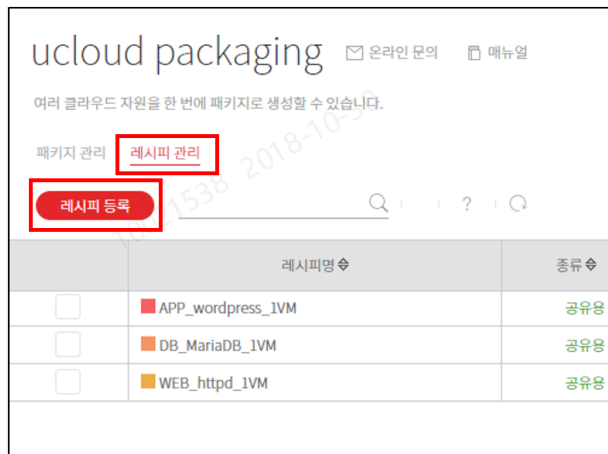
(주의) 패키지를 강제 삭제한 경우, 포함된 자원의 잔여 여부를 확인 바랍니다. 자원이 남아있어 원하지 않는 과금이 될 수 있습니다.

2.3.3 레시피 관리

레시피 작성법은 여기에서 별도로 안내합니다.

□ 레시피 등록

○ packaging > 레시피 관리 > 레시피 등록 을 클릭합니다.



○ 레시피명과 설명을 입력합니다.

○ 업로드 할 json 형식의 파일을 '파일선택'을 눌러 선택합니다.

- 업로드 후 '점검'을 클릭하여 올바른 json 형식인지 점검 받습니다.
- 만약 형식이 잘못된 경우 점검에서 오류가 발생합니다. 레시피에 잘못된 부분이 없는지 다시 검토 후 저장하여 업로드 합니다.

○ 점검이 완료되면 '생성'을 클릭하여 업로드 합니다.

(주의) U를 통한 업로드의 경우, 레시피 파일 크기가 20KB 이하로 제한됩니다.

□ 레시피 상세

ucloud packaging 📧 온라인 문의 📖 매뉴얼

여러 클라우드 자원을 한 번에 패키지로 생성할 수 있습니다.

패키지 관리 레시피 관리

레시피 등록 🔍 ? 🔄

☐	APP_wordpress_1VM	공유용
☒	DB_MariaDB_1VM	공유용
☐	WEB_httpd_1VM	공유용

✓ 선택된 템플릿: DB_MariaDB_1VM

상세정보 레시피

- 레시피명 DB_MariaDB_1VM
- 레시피 ID urn:ucloud:upac:-2142651461:DB_MariaDB_1VM:675af421-12a2-4ce6-b7fb-125113b30154
- 생성일시 2018-10-26 09:10:48

○ 상세정보 : 선택한 레시피의 이름, ID, 생성일시를 확인할 수 있습니다.

○ 레시피 : 선택한 레시피의 json 문서(Recipe Body)를 볼 수 있습니다.

[TIP] 공유용 레시피의 문서(Body)를 복사하여 필요에 따라 수정하는 방법이 개인용 레시피 작성에 매우 도움이 됩니다.

2.4 ucloud Packaging 레시피 제작

2.4.1 Packaging 레시피 소개

□ 개요

레시피는 Packaging에서 패키지를 배포 할 때 어떻게 만들것인가를 기술한 하나의 문서입니다. 사전에 어떻게 만들것인지에 대해 프로그래밍 하듯이 미리 JSON 문서로 만들어두고, 패키지 생성이 필요할 때 마다 사용하여 배포에 사용됩니다.

□ 레시피 언어 포맷

ucloud Packaging 레시피는 JSON 텍스트 파일입니다. 포맷은 JSON 표준을 따릅니다. 본 매뉴얼에서는 JSON에 대한 설명을 따로 수록하지는 않습니다.

JSON에 대한 자세한 내용은 <http://www.json.org> 를 참고하시기 바랍니다.

2.4.2 Packaging 레시피 구성

[TIP] 레시피를 처음부터 작성하는 것은 어려운 일입니다.

미리 작성되어 제공되는 '공유용 레시피'를 복사하여, 필요에 따라 수정하여 사용하는 것이 더욱 수월합니다.

레시피는 크게 아래 다섯 파트로 구성됩니다.

```

{
  "UPAC Template Version": "2018-01-01",
  "Description": "설명",
  "Parameters": {
    "Param1": {
      "Description": "Parameter 1을 입력하세요.",
      "Type": "String",
      "Default": "paramparamparam"
    },
    "size": {
      "Description": "크기를 입력하세요",
      "Type": "String",
      "Default": "80"
    }
  },
  "Resources": {
    "VM1": {
      "Type": "UPAC::VirtualMachine",
      "Properties": {
        "ZoneId": {
          "Ref": "Param1"
        }
      }
    },
    "Disk1": {
      "Type": "UPAC::Volume",
      "Properties": {
        "Size": {
          "Ref": "size"
        }
      }
    }
  },
  "Outputs": {
    "MyVMPassword": {
      "Value": {
        "GetAtt": [
          "VM1",
          "password"
        ]
      },
      "Description": "생성된 VM1의 SSH 비밀번호"
    }
  }
}

```

▣ Version (옵션)

레시피가 작성된 버전을 의미합니다. 작성된 날짜 형태로 입력됩니다.

▣ Description (옵션)

레시피 설명을 입력합니다. 레시피 본문(Body)에서 보는 용도이며, UI에서 보이는 레시피 설명과는 별도입니다. 길이는 제한은 4000자 이내입니다.

▣ Parameters (옵션)

레시피 작성 과정에서 지정하지 않고, 이 레시피로 패키지를 배포할 때 입력 받을 값(파라미터)를 정합니다. 예를 들어 동일한 레시피(구조)로 다른 Zone에 여러 번 배포하고 싶으면 Zone ID를 파라미터로 기술해 두면, 배포할 때마다 Zone을 다르게 해서 배포가 가능합니다.

파라미터로 지정된 값은 Resources나 Outputs에서 참조하여 사용할 수 있습니다.

형식은 아래를 따릅니다.

```
"Parameters" : {  
  "MyParam1" : {  
    "Type" : "xxxx",  
    "Default" : "yyyy",  
    (기타 다른 내용들)  
  },  
  "Another" : {  
    "Type" : "zzzz",  
    (기타 다른 내용들)  
  }  
}
```

각 파라미터 안에 들어갈 수 있는 내용들은 아래와 같습니다.

항목	필수여부	설명
Type	Yes	Boolean, String, Number, CommaDelimitedList 중 한 가지 타입을 부여합니다. String 타입은 문자열을 의미합니다. Number 타입은 정수나 소수점 숫자를 표현합니다. Number 타입에 대한 값은 integer 나 float 일 수 있지만, Number 의 값은 String 타입과 같이 CommaDelimitedList 타입은 쉼표(,)로 분리된 String 타입의 배열입니다. 배열에 포함되는 String 의 값에는 공백이나 쉼표가 들어가는는 안된다.
Default	No	package 생성 시 사용자가 parameter 값을 지정하지 않을 경우 사용되는 기본 값입니다. 만약 Default 항목이 없는 Parameter 에 대해 생성 시 값이 전달되지 않으면 패키지 생성이 실패 Default에 기입되는 값 역시 parameter 값의 제약(아래)을 지켜야 합니다.
NoEcho	No	만약 "TRUE"일 경우에 별표(***)로 값이 치환되어 표시됩니다.
AllowedValues	No	parameter 값으로 허락된 값들의 배열입니다.
AllowedPattern	No	String 타입에 대한 제약사항입니다. parameter 의 string 타입 값에 허용되는 패턴을 정규 표현
MaxLength	No	String 타입에 대한 제약사항입니다. parameter 의 string 타입 값의 최대 길이를 정합니다.
MinLength	No	String 타입에 대한 제약사항입니다. parameter의 string 타입 값의 최소 길이를 정합니다.
MaxValue	No	Number 타입에 대한 제약사항입니다. parameter 의 number 타입 값의 최대 값을 정합니다.
MinValue	No	Number 타입에 대한 제약사항입니다. parameter 의 number 타입 값의 최소 값을 정합니다.
Description	No	parameter 에 대한 4000 자 이내의 설명입니다.

▣ Resources (필수)

패키지에 배포할 자원들을 정의합니다.

자원은 일반적으로 생각할 수 있는 VM, Disk, WAF, DB, NAS 뿐 아니라 네트워크 연결 작업이나 서버 추가 작업 등 작업 자체도 자원으로 포함됩니다.

예를 들면, 포트포워딩 작업, 로드밸런서에 서버 추가, 오토스케일링 그룹 설정 등이 포함됩니다.

Resources 항목은 크게 Type, Properties, DependsOn 으로 구분 됩니다.

- Type : 자원의 유형을 의미합니다. VM은 VirtualMachine 이라는 타입이고, 포트포워딩 룰은 PortForwardingRule이라는 타입입니다.
- Properties : 각 타입의 자원을 만들기 위해 필요한 속성을 입력합니다. 예를 들면, VM 타입의 자원은 사양, OS, Zone, userdata 등의 속성들을 가집니다.
- DependsOn : 자원의 배포 순서를 지정하기 위해 사용하는 항목입니다.

대표적인 Resources 중 하나인 VM의 정의 예시 입니다.

```
"Resources" : {
  "MyVM" : {
    "Type" : "UPAC::VirtualMachine",
    "Properties" : {
      "ProductCode": "std_cent 7.0 64bit en_1x1_roortonly",
      "DisplayName": "MyServer",
      "ZoneId": "qqqq-wwwwwww-rrrrrrr"
    }
  }
}
```

Type으로 VirtualMachine을 지정하고 있고, 그에따라 필요한 Properties인 서비스오퍼링, VM 이름, VM 이미지, Zone 등을 정하고 있습니다.

Resources의 종류별 Type과 그에 따른 Properties는 내용이 많아 별도 페이지에서 설명합니다. 여기를 참고하세요.

▣ Outputs (옵션)

이 레시피를 사용하여 패키지 배포가 끝난 후, 사용자에게 반환해 줄 결과값을 지정합니다. 예를 들어, VM을 배포한 후 그 VM의 비밀번호를 전달할 수 있습니다.

2.4.3 레시피 내장 함수

레시피 내에서 사용할 수 있는 내장 함수를 소개합니다.

▣ Ref

특정 파라미터나 자원의 어떤 값을 참조할 때 사용합니다.

○ 사용 방법

```
"Ref" : "논리적 이름"
```

논리적 이름에는 파라미터명 혹은 자원의 논리적 이름을 사용할 수 있습니다.

파라미터명을 명시하면, 패키지를 배포할 시점에 배포자가 해당 파라미터로 입력한 값을 그대로 사용합니다.

자원의 논리적 이름을 사용할 경우, 해당 자원에 정해진 값을 반환합니다. 정해진 값은 아래와 같습니다.

자원 유형	반환되는 항목
VirtualMachine	id
Volume	id

VolumeAttachment	(반환값 없음)
FirewallRule	id
IpAddress	id
PortForwardingRule	id
LoadBalancerRule	id
LoadBalancerAssignment	id
LoadBalancer	Id(loadbalancerid)
LoadBalancerWebServerAddition	id(serviceid)
WAF	id
WAFWebServerAddition	id(webserverid)
WAFWebSiteAddition	id(websiteresourceid)
DB	id(instanceid)
DBParameterGroup	id(parameter groupid)
DBAccessControlGroup	id(accesscontrol groupid)
DBReplicationGroup	id(replication groupid)
DBReplicationGroupMultiZone	id(replication groupid)
DBHaGroupSingleZone	id(ha groupid)
AutoScalingGroup	(반환값 없음)
LaunchConfiguration	(반환값 없음)
ScalingPolicy	Id(scalingpolicyurn)
Topic	Id(topicurn)
Alarm	(반환값 없음)
WAIT_OBJECT	UserSignal
WAIT_OBJECT_HANDLE	id

○ 사용 예시 1 (파라미터 반환)

```
"Parameters": {
  "zone1": {
    "Type": "String"
  }
}
```



```

},
"Resources": {
  "MyVM": {
    "Type": "UPAC::VirtualMachine",
    "Properties": {
      "ZoneID": {
        "Ref": "zone1"
      }
    }
  }
}
}

```

Parameters에서 zone1이라는 파라미터를 입력받아서, 실제로 Resources 내 MyVM이라는 VirtualMachine을 만들 때 ZoneID로 사용합니다.

○ 사용 예시 2 (자원 반환)

```

"Resources": {
  "MyVM": {
    "Type": "UPAC::VirtualMachine"
  },
  "MyIP": {
    "Type": "UPAC::IpAddress"
  },
  "DoPortForwarding": {
    "Type": "UPAC::PortForwardingRule",
    "Properties": {
      "VirtualMachineId": {
        "Ref": "MyVM"
      },
      "IpAddressId": {
        "Ref": "MyIP"
      },
      "PublicPort": "22",
      "PrivatePort": "22",
      "Protocol": "TCP"
    }
  }
}
}

```

▣ GetAtt

입력 문자를 Base64로 인코딩 합니다. VM 생성할 때 userdata 전송할 때 주로 사용됩니다.

○ 사용 방법

```

"GetAtt": [
  "자원 이름",
  "속성 이름"
]

```

자원 별 속성은 아래와 같습니다.

자원 유형	속성 이름
VirtualMachine	IpAddress, password
Volume	
VolumeAttachment	
FirewallRule	
IpAddress	IpAddress

PortForwardingRule	
LoadBalancerRule	
LoadBalancerAssignment	
LoadBalancer	LoadBalancerId, ServiceIp
LoadBalancerWebServerAddition	
WAF	ServiceIp
WAFWebServerAddition	
WAFWebSiteAddition	
DB	Port, Ip, PrivatPort, PrivateIp
DBParameterGroup	
DBAccessControlGroup	
DBReplicationGroup	
DBReplicationGroupMultiZone	
DBHaGroup	havip
AutoScalingGroup	
LaunchConfiguration	
ScalingPolicy	
Topic	
Alarm	
WaitObject	
WaitObjecthandle	

o 사용 예시

```

"Outputs": {
  "MyPassword": {
    "Value": {
      "GetAtt": [
        "MyVM",
        "password"
      ]
    },
    "Description": "VM 비밀번호"
  }
}

```

□ Base64

입력 문자를 Base64로 인코딩 합니다. VM 생성할 때 userdata 전송할 때 주로 사용됩니다.

○ 사용 방법

```
"Base64" : {인코딩할문자열}
```

○ 사용 예시

```
"MyVM": {
  "Type": "UPAC::VirtualMachine",
  "Properties": {
    "UserData": {
      "Base64": {
        "yum install -y httpd"
      }
    }
  }
}
```

□ Join

원하는 값들을 구분자로 연결된 하나의 문자열로 만들 때 사용합니다. 구분자가 빈 문자열인 경우, 모든 값들이 구분자 없이 연결 됩니다.

○ 사용 방법

```
"Join": [
  "구분자", [
    쉼표(,)로 구분된 값들의 목록
  ]
]
```

○ 사용 예시 1

```
"Join": [
  ":", [
    "a",
    "b",
    "c"
  ]
]
```

이 함수의 실행 결과는 "a:b:c" 가 된다.

○ 사용 예시 2 (VM 생성 시 user data로 여러 줄 전달)

```
"UserData": {
  "Base64": {
    "Join": [
      "", [
        "yum install -y httpdWn",
        "systemctl start httpdWn",
        "systemctl enable httpdWn"
      ]
    ]
  }
}
```

세 줄의 커맨드라인이 하나의 명령어가 되어 Base64 인코딩 후 user data로 전달 됩니다.

