OpenInfra DAYS Korea

OVN 환경의 OpenStack 방화벽

FWaaS와 보안 그룹 동작 분석

김영수 (kt cloud Foundation플랫폼 팀)

kt cloud

OpenInfra DAYS Korea

kt cloud

kt cloud CLOUD

No.1 클라우드 사업자

국내 최초 Public Cloud. 다수의 고객사가 신뢰하는 디지털 혁신 파트너

핵심역량

공공/금융 특화 전문성:
국내 1호 클라우드 보안인증(CSAP) 획득, 규제를 준수하는 맞춤형 금융 클라우드 제공

검증된 기술력과 경험:
모든 산업군에 걸친 고객 레퍼런스 기반의 최적화된 솔루션 제공

유연한 확장성:
Multi/Hybrid Cloud 환경을 손쉽게 구성하고 글로벌 CSP와 직결 연동 지원

주요서비스

Public / Private Cloud, 공공 / 금융 전용 클라우드, 5G Edge Cloud 등

kt cloud DC

No.1 데이터센터 리더

20여 년의 노하우, AI 기반으로 운용되는 가장 안전하고 안정적인 IT 인프라

핵심역량

압도적인 규모와 입지:
수도권 9개를 포함한 전국 15개 최다 데이터센터 보유, 고객 맞춤 위치 제공

글로벌 수준의 운용 역량:
20년 이상의 운용 경험과 AI 기반 자동화 관제로 글로벌 Top 수준의 품질 제공

국내 유일 통합 제공:
데이터센터(DC), 클라우드, 네트워크(NW)를 직접 보유하고 통합 컨설팅 및 MSP 제공

주요서비스

Colocation, MSP, DBO, DCO, 인프라 진단 서비스 등

kt cloud

OpenInfra DAYS Korea

목차

kt cloud

OpenInfra DAYS Korea

1부

서론 - 어떻게 시작되었나?

kt cloud

OpenInfra DAYS Korea

kt cloud's

NEXT Project

kt cloud 차세대 플랫폼 'NEXT'

기존 아키텍처 전환

Classic 이라 호칭되는 기존의 H/W 기반 네트워크 가상화와 벤더 솔루션 기반으로 구성된 기존 아키텍처에서 탈피

신규 표준 환경 구축

S/W 기반의 네트워크 가상화와 오픈소스 기반의 신규 아키텍처 등을 포함한 새로운 표준 클라우드 환경 수립 및 구축

핵심 기술

Kubernetes, Kube-OVN, **OVN**, OpenStack

적용 범위

신규 클라우드 존 에 적용 예정

kt cloud

OpenStack API 적극 활용

네이티브 API 활용

OpenStack이 제공하는 네이티브 API를 최대한 활용하여 안정성과 호환성을 확보합니다.

기술 내재화

API 기술을 깊이 이해하고 내재화하여 효율적인 시스템 구축과 운영을 실현합니다.

글로벌 오픈소스에 기여

오픈소스 커뮤니티에 적극 기여하여 글로벌 생태계 발전에 참여하고 기술 혁신을 주도합니다.

→ 핵심 기술 역량 확보 및 오픈소스 생태계 주도 참여

NACL 기능 구현을 위한 FWaaS 분석

기능 구현 필요

NACL(Network Access Control List) 기능 요구사항 발생

기술 조사

OpenStack Project 조사 → Neutron FWaaS 발견

FWaaS 선택

"OpenStack이 제공하는 기능을 활용하자"

FWaaS 분석 및 테스트

FWaaS 분석 및 테스트 시작

Security Group과 FWaaS의 연관 관계 분석

두 방화벽 시스템 간의 상호작용과 우선순위 관계 분석

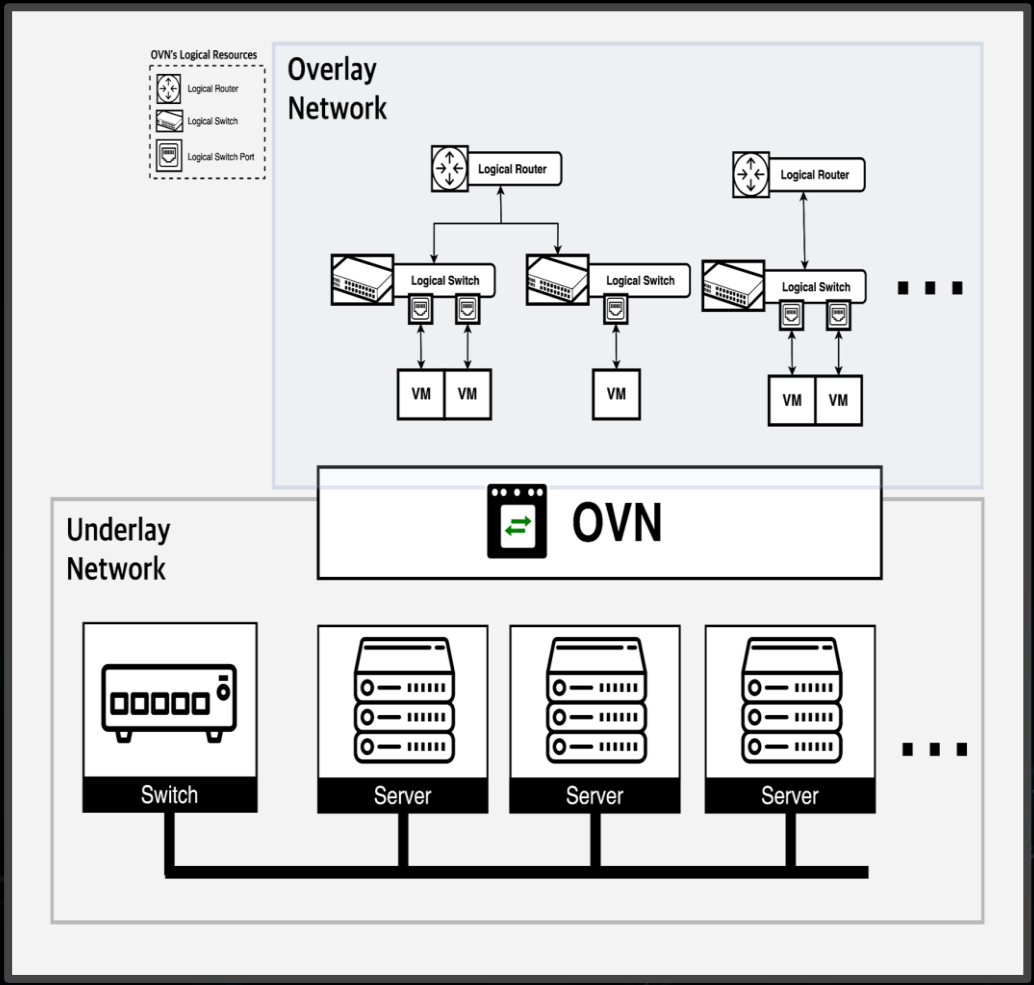
OpenInfra DAYS Korea

2부

OVN (Open Virtual Network)

kt cloud

OVN (Open Virtual Network)



OVN은 무엇인가요?

OVN은 Open vSwitch의 기능을 확장하여 가상 네트워크를 쉽게 생성하고 관리할 수 있도록 지원합니다. 이는 클라우드에서 빠르게 변화하는 네트워크 요구 사항을 충족하는 데 필수적입니다.

주요 장점

중앙 관리: OVN-Northd를 통해 네트워크 규칙을 한곳에서 일관되게 적용합니다.

빠른 처리: OVS가 데이터를 빠르고 효율적으로 처리합니다.

확장성: 대규모 클라우드 환경에 적합하며, 여러 오버레이 기술(VXLAN, Geneve 등)을 지원합니다.

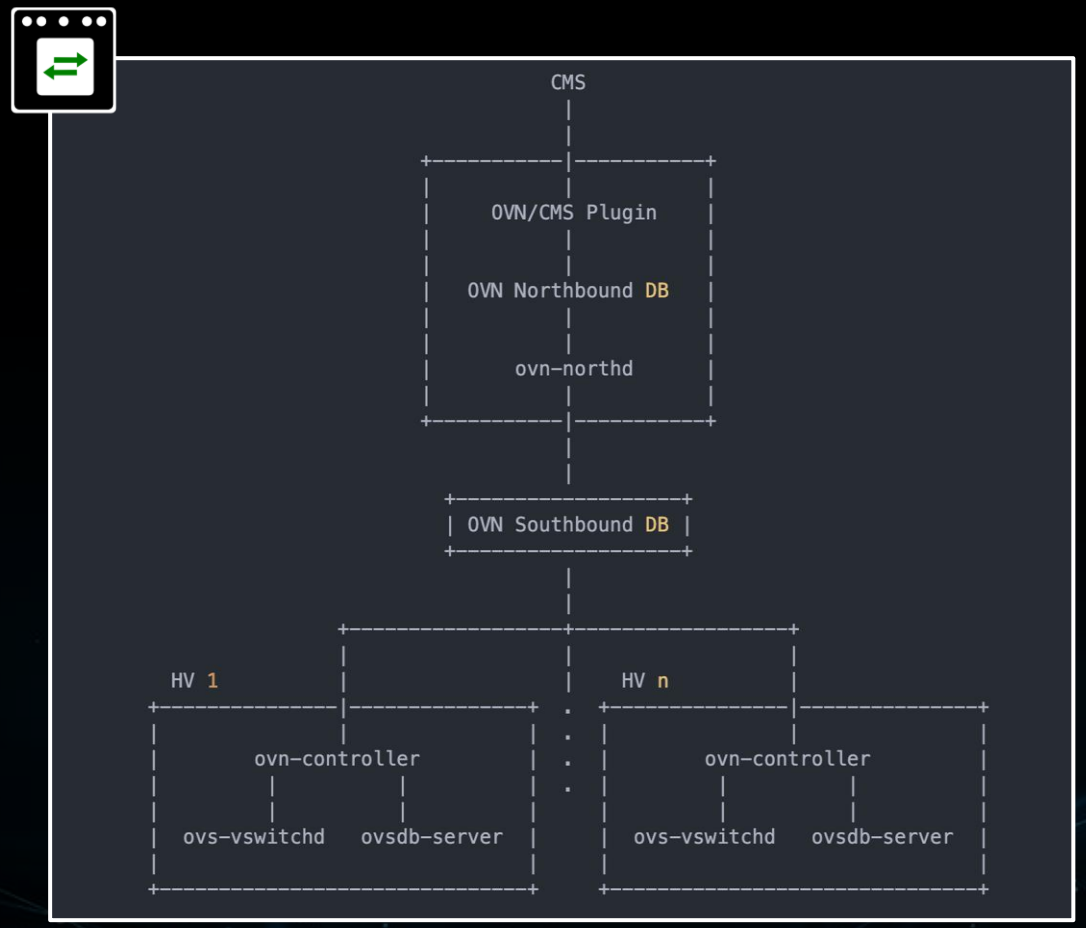
보안: 네트워크 분리 및 보안 그룹 설정 기능을 제공합니다.

OpenStack과의 연동

openstack.

주로 OpenStack Neutron의 핵심 기술로 사용됩니다. OpenStack 환경에서 가상 머신 간의 네트워크 연결 및 규칙 관리를 자동화하며, Nova, Cinder와 같은 다른 OpenStack 서비스와도 잘 연동됩니다.

OVN 아키텍처 한눈에 보기



중앙 관리 계층

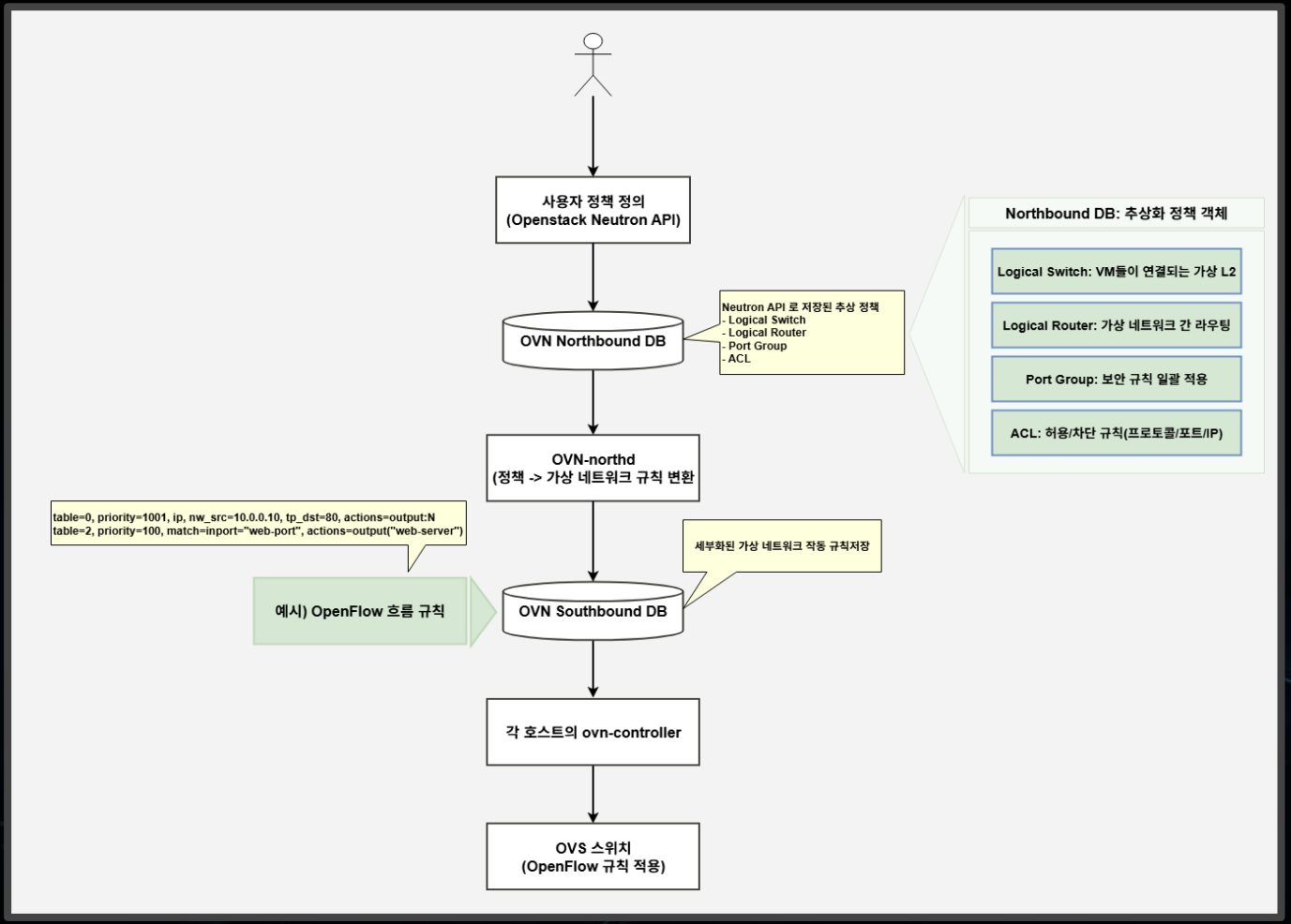
- CMS: OpenStack, Kubernetes 등
- OVN/CMS Plugin: Neutron 플러그인 등 CMS별 인터페이스
- OVN Northbound DB: 논리적 네트워크 설정 저장
- ovn-northd: NB DB → SB DB 변환

분산 실행 계층

- OVN Southbound DB: 논리 플로우, 물리 정보 저장
- ovn-controller: 각 하이퍼바이저의 OVN 에이전트
- ovs-vswitchd: 실제 스위칭 수행
- ovsdb-server: 로컬 OVS 설정 관리

중앙 집중식 제어, 분산 실행, 선언적 설정

네트워크 정책이 “논리적 흐름” 으로 변환되는 과정



OVN 네트워크 정책 처리 과정

☰ 1단계: 사용자 설정 저장

Neutron API → Northbound DB

- 사용자 네트워크 설정 입력 (가상 스위치, 라우터, ACL 등)
- 상위 레벨 네트워크 정책으로 저장

🐙 2단계: 정책 변환

ovn-northd 엔진

- 복잡한 네트워크 정책 → 세부 OpenFlow 지시문
- 가상 네트워크 작동 규칙으로 변환

☲ 3단계: 실행 준비

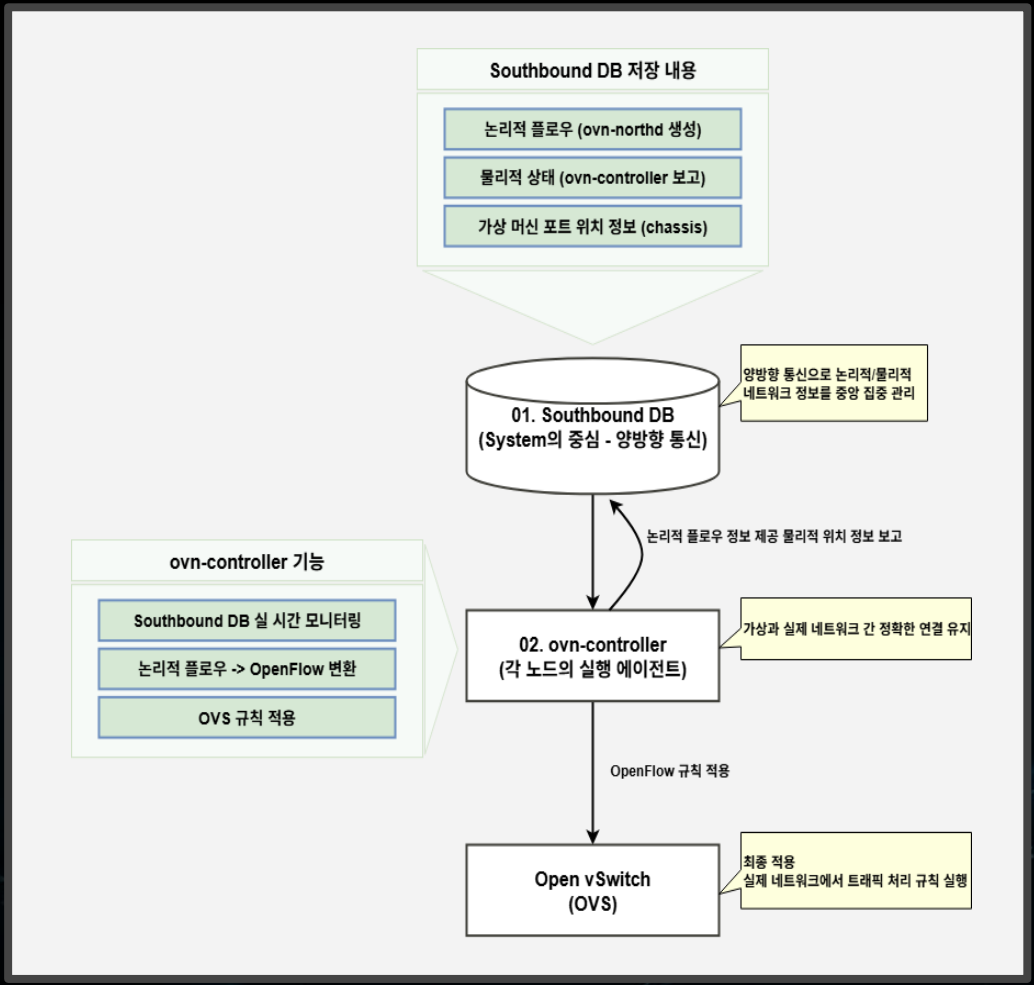
Southbound DB

- 변환된 논리적 흐름을 각 서버에 배포
- OVS 스위치에 실제 적용

사용자 친화적 설정 → 네트워크 장비 실행 지시문

kt cloud

논리적 흐름이 “실제 네트워크”로 구현되는 과정



Southbound DB & ovn-controller

☰ Southbound DB: 중앙 집중식 저장소

- 논리적 플로우 저장 (ovn-northd 생성 규칙)
- 물리적 상태 저장 (각 노드 실제 네트워크 상태)
- 실시간 구독 지원으로 정보 배포

⚙ ovn-controller: 각 노드 실행 에이전트

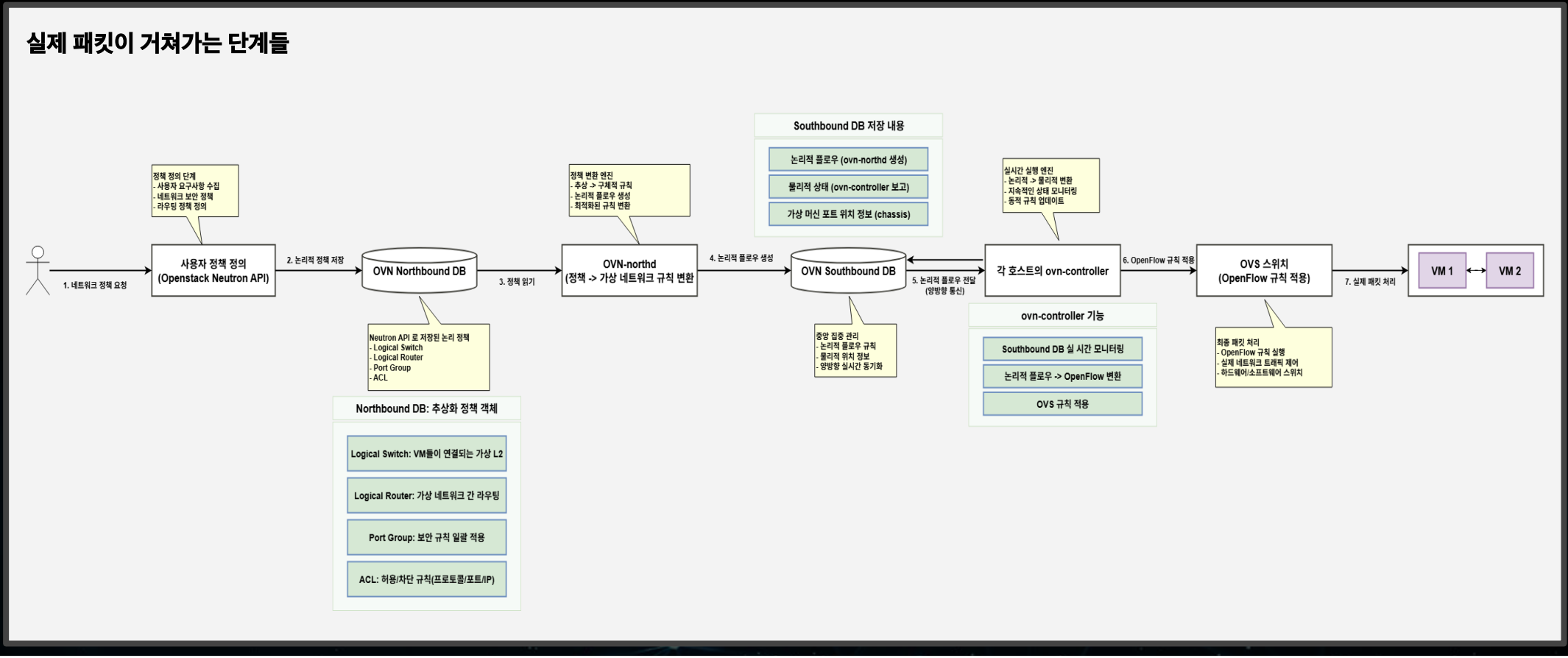
- SB DB 실시간 모니터링 (Pull 방식)
- 논리적 플로우 → OpenFlow 규칙 변환
- VM 포트 물리적 위치(Chassis) 정보 보고

↻ 동작 과정

1. 읽기: SB DB에서 해당 서버 관련 논리적 플로우 획득
2. 변환: OpenFlow 규칙으로 변환 처리
3. 적용: 실제 OVS에 규칙 적용
4. 보고: 물리적 네트워크 상태 변화를 SB DB에 보고

중앙 저장소(SB DB) + 분산 실행 에이전트(ovn-controller) = 가상 네트워크 구현

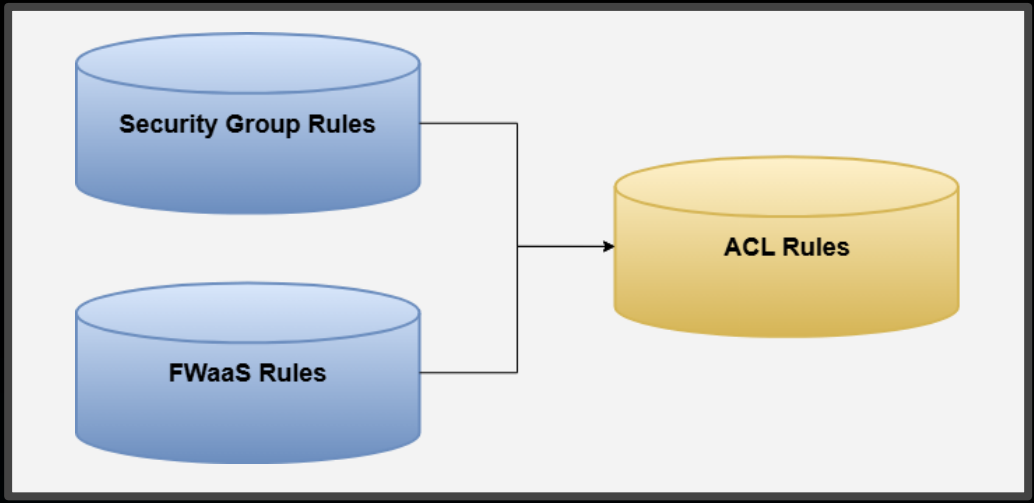
OVN 패킷 처리 흐름



사용자 네트워크 설정(Neutron API) → 논리적 정책 저장(Northbound DB) → 실행 규칙 변환(ovn-northd → Southbound DB) → 물리적 네트워크 구현(ovn-controller → OVS) → VM 간 실제 통신

OpenInfra DAYS Korea

OVN ACL (Access Control List)



실제 ACL 예시

```
_uuid               : 4be04b2c-eaf3-49fb-8484-628e7346682d
action              : allow-related
direction           : from-lport
external_ids        : {"neutron:security_group_rule_id"="3ad05a85-f11a-4a97-bd0b-dbafd0a52c06"}
label               : 0
log                 : false
match               : "inport == @pg_0f1e7d8e_df31_4bfb_ba55_7dfd50aa3e0a && ip4"
meter               : []
name                : []
options             : {}
priority            : 1002
severity            : []
tier                : 0

_uuid               : 3b697ce6-b744-4fb2-a5b6-c61dda89b32e
action              : allow-stateless
direction           : to-lport
external_ids        : {"neutron:firewall_rule_id"="e0b2e107-858b-4f44-a9f2-696d61b4ad21"}
label               : 0
log                 : false
match               : "outport == @pg_733120f4_2481_4342_9110_43ddbaabf855 && ip6"
meter               : []
name                : []
options             : {}
priority            : 1999
severity            : []
tier                : 0
```

1 ACL의 정의와 역할

ACL (Access Control List)은 네트워크 트래픽을 제어하는 규칙 집합.

2 ACL의 주요 구성 요소

- **match**: 패킷이 이 ACL 규칙에 일치하는지 여부를 결정하는 조건입니다. (예: `ip4.src == 192.168.1.0/24 && tcp.dst == 80`)
- **action**: 규칙에 일치하는 패킷에 대해 수행할 작업. (`allow`, `drop`, `reject` 등)
- **priority**: 여러 ACL 규칙이 충돌할 경우 우선순위를 결정. 높은 숫자가 높은 우선순위를 가지게 됨.
- **direction**: ACL이 적용될 트래픽의 방향입니다. (`from-lport` (Egress), `to-lport` (Ingress))

3 Security Group 및 FWaaS의 ACL 변환

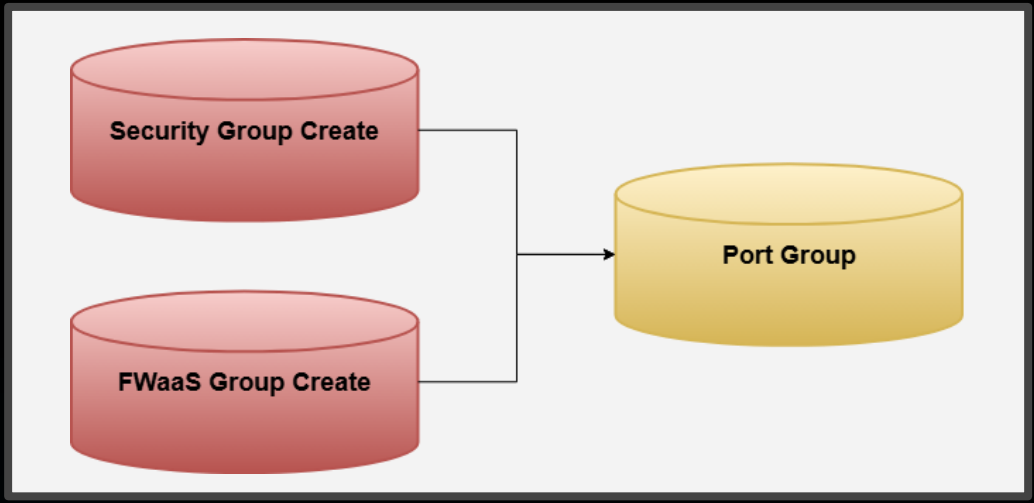
OpenStack의 Security Group 및 FWaaS 규칙은 OVN 내부적으로 ACL 규칙으로 변환.
이 변환을 통해 가상 머신 포트나 논리 라우터에 세분화된 트래픽 제어가 적용 됨.

4 ACL 적용 위치

- **Logical Switch**: Security Group 규칙이 적용.
- **Logical Router**: FWaaS 규칙이 적용.

kt cloud

OVN Port Group



실제 Port Group 예시

```
_uuid               : b842470c-ad8b-44ce-acb0-cbed5d905265
acls                : [2fb22ccf-9010-4111-9abb-351deeb17be0]
external_ids        : {"neutron:security_group_id"="ff3695cd-47ab-4ff8-8249-9c3ad6bffb5a"}
name                : pg_ff3695cd_47ab_4ff8_8249_9c3ad6bffb5a
ports               : [002b9d5b-986a-4b17-880c-a5737426e63d, 6799fd23-688f-493d-bfb0-bb7d8c178dbb,

_uuid               : 04afbd89-8bca-4689-82b1-42d3367d0005
acls                : [4046c8f1-6c67-4f71-a461-78f1b69837c5, 688eb984-9b40-46d0-b81c-f2c03cacbf4c,
1929a4dd]
external_ids        : {"neutron:firewall_group_id"="c92c453f-60cf-4351-9922-5a8e870fb0be"}
name                : pg_c92c453f_60cf_4351_9922_5a8e870fb0be
ports               : [dc8832a1-ec42-4310-9958-45d62c001de0]
```

1 Port Group의 정의와 목적

Port Group은 OVN 내에서 논리 포트들의 집합. 여러 포트에 동일한 접근 제어 규칙(ACL)을 적용할 수 있어, ACL 관리 단순화와 효율성 향상이 목적.

2 Port Group의 주요 특징과 장점

- 동적 할당: 그룹에 속하는 포트는 동적으로 추가,삭제 가능. 이는 ACL 규칙에 자동으로 반영 됨.
- ACL 규칙 단순화: 포트가 아닌 그룹에 ACL을 적용하여 규칙의 수를 줄이고 가독성을 향상.
- 관리 용이성: 대규모 환경에서 수많은 논리 포트를 효율적으로 관리.

3 Openstack 과 Port Group의 관계

OpenStack 환경에서 생성되는 Security 및 FWaaS Group 는 OVN Port Group과 ACL로 변환. Security 및 FWaaS Group은 각 OVN Port Group에 매핑되며, 해당 Group 에 속하는 가상 머신 포트들은 해당 Port Group의 멤버가 됨.

4 Port Group 생성 및 관리 방법

- OpenStack과 같은 상위 컨트롤러가 보안 그룹 규칙을 생성하면, Neutron ML2 OVN Driver가 이를 Port Group 및 ACL 규칙으로 자동 변환하여 OVN NB DB에 기록. 이후 ovn-northd가 NB의 정책을 SB의 논리 플로우로 변환."

OpenInfra DAYS Korea

3부

OpenStack의 두 가지 방화벽

Security Group과 FWaaS 이해

kt cloud

OpenInfra DAYS Korea

OpenStack Security Group

Security Group 은 클라우드 환경에서 가상 머신(VM) 간의 트래픽을 제어하는 Stateful 방화벽 기능.

VM 단위 방화벽
일반적으로 각 가상 머신(VM)의 가상 NIC(vNIC)에 적용되어, 개별 인스턴스 수준에서 트래픽 제어.

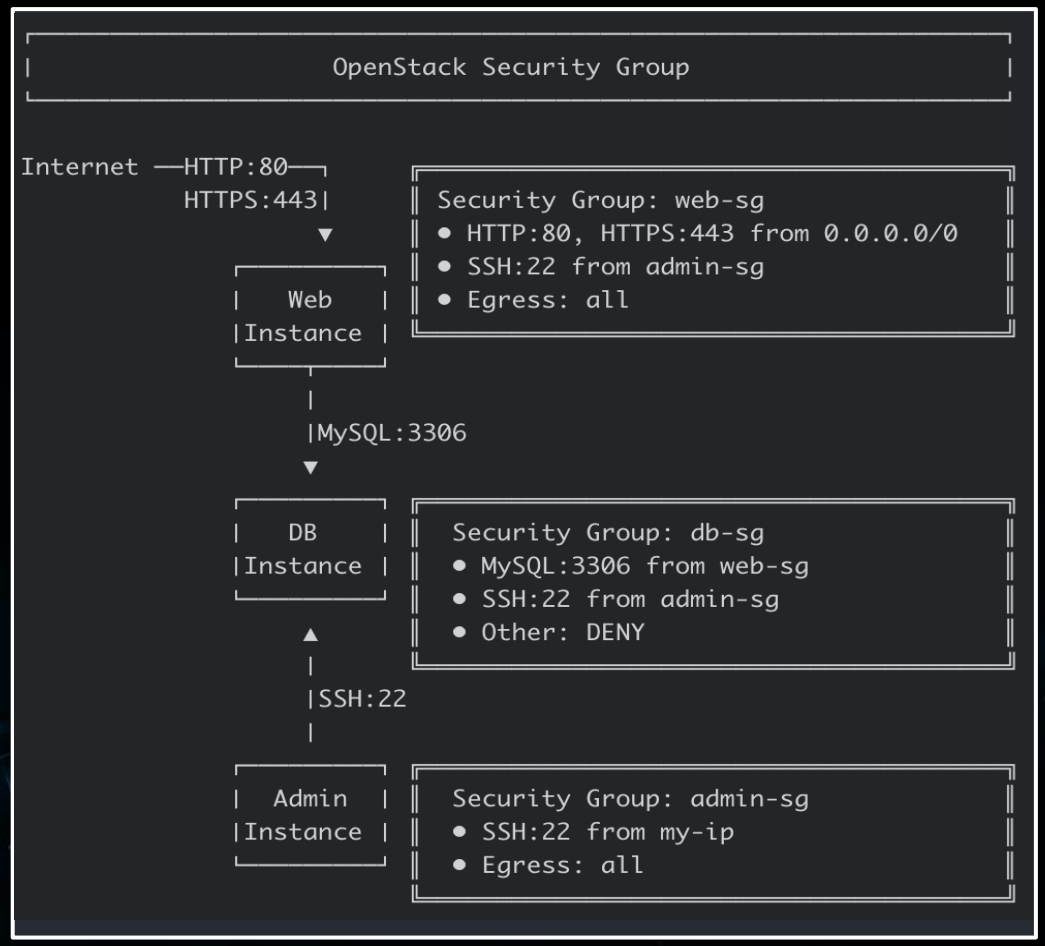
포트별 세밀한 제어
특정 프로토콜(TCP, UDP, ICMP), 포트 번호, 그리고 소스 IP 주소 범위(CIDR)에 따라 인바운드 및 아웃바운드 허용 트래픽 제어.

인스턴스 단위 규칙 적용
클라우드 환경에서 생성되는 각 인스턴스에 따라 맞춤형 방화벽 규칙을 유연하게 적용할 수 있어, 개별 서비스의 보안 요구사항을 충족.

양방향 규칙 설정
특정 IP 대역에서만 SSH(TCP/22) 접속을 허용하거나, 모든 아웃바운드 트래픽을 허용하는 등 인바운드와 아웃바운드 트래픽에 대한 규칙을 독립적으로 설정 가능.

동적 업데이트 및 적용
규칙 변경 시, 해당 변경 사항이 즉시 또는 매우 짧은 시간 내에 관련 VM에 적용. 이는 서비스 중단 없이 보안 정책을 유연하게 조정 가능.

"인스턴스에 적용되는 가상 방화벽으로, 필요한 것만 열어주는 간단하고 직관적인 접근 제어 방식"



kt cloud

OpenInfra DAYS Korea

Security Group 은 OVN 에서 어떻게 구현 될까?

Neutron Security Group → OVN Port Group + ACL 변환

Neutron Security Group Create

```
# openstack security group create <SG Name>
```

```
+-----------------+-----------------------------------------------------------------------------------------------+
| Field           | Value                                                                                         |
+-----------------+-----------------------------------------------------------------------------------------------+
| created_at      | 2025-08-20T07:42:53Z                                                                          |
| description     | web-sg                                                                                        |
| id              | b9158ef8-774d-4e9b-9d55-2934f8920f2d                                                          |
| name            | web-sg                                                                                        |
| project_id      | 9745b7b8dc9942e3856ce9d321ad0fd5                                                              |
| revision_number | 1                                                                                             |
| rules           | created_at='2025-08-20T07:42:53Z', direction='egress', ethertype='IPv4',                      |
|                 | id='4335b973-7c1e-4711-8d79-c83b3873db3a', standard_attr_id='289',                            |
|                 | updated_at='2025-08-20T07:42:53Z'                                                             |
|                 | created_at='2025-08-20T07:42:53Z', direction='egress', ethertype='IPv6',                      |
|                 | id='63df0de9-8bf0-4c75-bb51-264a06c44307', standard_attr_id='290',                            |
|                 | updated_at='2025-08-20T07:42:53Z'                                                             |
| shared          | False                                                                                         |
| stateful        | True                                                                                          |
| tags            | []                                                                                            |
| updated_at      | 2025-08-20T07:42:53Z                                                                          |
+-----------------+-----------------------------------------------------------------------------------------------+
```

Security Group 을 생성하면 Default Rule 로 인해 Port Group, ACL 이 생성.

적용

적용

Port Group

```
# ovn-nbctl list port-group
```

```
_uuid               : 0cfd553b-efb7-4bfd-9c32-27cd830aea53
acls                : [46b1e493-6302-4ac3-ab60-a5aef9808011, bf9aff4e-d1f6-4632-889f-cc78751a1774]
external_ids        : {"neutron:security_group_id"="b9158ef8-774d-4e9b-9d55-2934f8920f2d"}
name                : pg_b9158ef8_774d_4e9b_9d55_2934f8920f2d
ports               : []
```

ACL Rule

```
# ovn-nbctl list acl
```

```
_uuid               : 46b1e493-6302-4ac3-ab60-a5aef9808011
action              : allow-related
direction           : from-lport
external_ids        : {"neutron:security_group_rule_id"="5f9a8074-b4a1-44ad-a20e-0f51fae2b46e"}
label               : 0
log                 : false
match               : "inport == @pg_b9158ef8_774d_4e9b_9d55_2934f8920f2d && ip6"
meter               : []
name                : []
options             : {}
priority            : 1002
severity            : []
tier                : 0
```

kt cloud

OpenInfra DAYS Korea

Security Group 의 Default Drop Group

Neutron 시작 시 neutron_pg_drop 이라는 Port Group 과 양방향 drop 규칙이 생성.

Neutron Security Group Create

```
_uuid               : 5b23fad9-4794-4761-8e65-e17c39509877
acls                : [a3f997f0-f7a7-42c8-b35f-2d4910eef869,
abb58451-0212-457d-bb2b-3ccb62c05a75]
external_ids        : {}
name                : neutron_pg_drop
ports               : [6799fd23-688f-493d-bfb0-bb7d8c178dbb]
```

```
_uuid               : a3f997f0-f7a7-42c8-b35f-2d4910eef869
action              : drop
direction           : from-lport
match               : "inport == @neutron_pg_drop && ip"
name                : []
priority            : 1001

_uuid               : abb58451-0212-457d-bb2b-3ccb62c05a75
action              : drop
direction           : to-lport
match               : "outport == @neutron_pg_drop && ip"
priority            : 1001
```

Security Group 규칙 외 모든 트래픽 차단

실제 적용 시나리오

시나리오 1: 일반 인스턴스 포트
Port Security: ✅ Enabled
Security Groups: web-sg, admin-sg
결과: Security Group + neutron_pg_drop 동시 적용

시나리오 2: 네트워크 장비 포트
Port Security: ❌ Disabled
Security Groups: None
결과: neutron_pg_drop에서 제외 (모든 트래픽 허용)

시나리오 3: 신뢰 포트 (DHCP, 라우터 등)
Trusted Port: ✅ True
결과: neutron_pg_drop에서 제외

주요 특징

특징	설명
자동 생성	Server 시작 시 자동으로 생성됨
시스템 레벨	사용자가 직접 관리하지 않는 시스템 Port Group
이중 보안	Security Group과 함께 작동하는 추가 보안 계층
정책	SG 규칙 외 모든 트래픽 차단
성능 최적화	단일 Port Group으로 모든 보안 포트 관리

kt cloud

Security Group 규칙의 실제 OpenFlow 변환

security group rule 생성은 security group 지정이 필요.

Openstack Security Group Rule 생성

```
# openstack security group rule create --proto tcp
--dst-port 80 --remote-ip 192.168.100.0/24 --ingress <SG Name>
```

Security group rule 의 priority 는 1002 로 지정.

OVN ACL 변환

```
_uuid               : a8870699-ac68-4439-811d-b97fe773c894
action              : allow-related
direction           : to-lport
external_ids        : {"neutron:security_group_rule_id"="<SG rule ID>"}
label               : 0
log                 : false
match               : "outport == @pg_<SG ID> && ip4 && ip4.src == 192.1
68.100.0/24 && tcp && tcp.dst == 80"
meter               : []
name                : []
options             : {}
priority            : 1002
severity            : []
tier                : 0
```

OpenFlow 최종 규칙

```
table=4 (ls_out_acl_eval    ), priority=2002 , match=(reg0[7] ==
1 && (outport == @pg_b9158ef8_774d_4e9b_9d55_2934f8920f2d && ip4
&& ip4.src == 192.168.100.0/24 && tcp && tcp.dst == 80)), action=
(reg8[16] = 1; reg0[1] = 1; next;)
```

예약된 시스템 우선 순위와의 충돌을 방지하기 위해 ACL OFFSET 1000을 추가하여 동작.

*1002 + **1000** = 2002*
ACL + OFFSET = Openflow

OpenInfra DAYS Korea

OpenStack Firewall as a Service

FWaaS는 클라우드 인프라의 중요한 보안 구성 요소로, 개별 가상 머신(VM)을 넘어 네트워크 경계에서 포괄적인 보안을 제공.

라우터 단위 방화벽

각 가상 머신이 아닌, 가상 라우터 인터페이스에 직접 적용되어 해당 네트워크 구간을 통과하는 모든 트래픽을 제어. 외부 공격으로부터 내부 네트워크를 보호하는 네트워크 게이트웨이 수준의 보안을 제공.

중앙 집중식 정책 관리

하나의 방화벽 정책 세트를 정의하고 이를 여러 라우터에 일관되게 적용할 수 있어, 복잡한 네트워크 환경에서도 효율적인 보안 관리 가능.

정교한 트래픽 필터링

소스/목적지 IP 주소, 프로토콜(TCP, UDP, ICMP 등), 포트 번호 등을 기반으로 인바운드 및 아웃바운드 트래픽을 허용하거나 차단 가능.

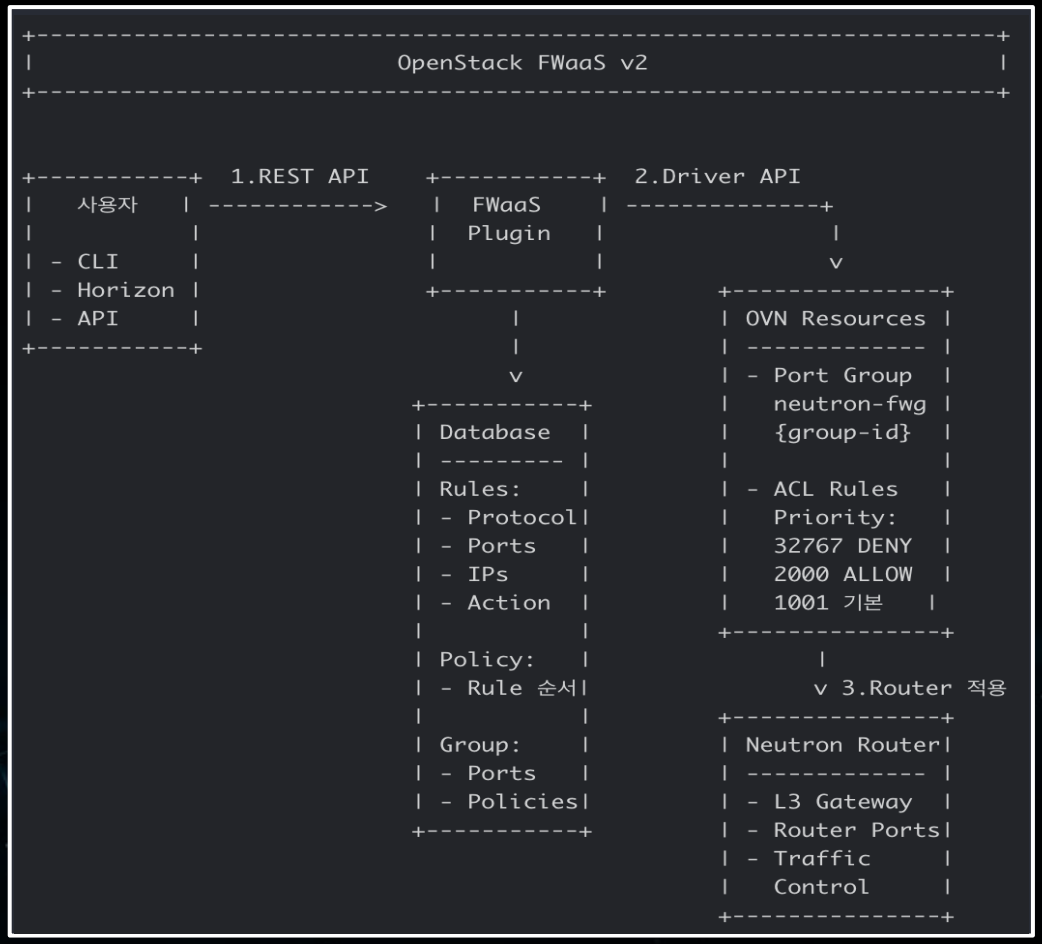
네트워크 세그멘테이션 강화

OpenStack FWaaS는 가상 라우터에 방화벽을 적용하여 그 관문을 지나는 네트워크 구역(서브넷) 간의 트래픽을 통제하는 중앙 보안 기능입니다.

동적 업데이트 및 적용

규칙 변경 시, 해당 변경 사항이 즉시 또는 매우 짧은 시간 내에 관련 VM에 적용. 이는 서비스 중단 없이 보안 정책을 유연하게 조정 가능.

"라우터에 적용되는 중앙 집중형 방화벽으로, 다양한 조건을 조합하여 허용과 차단을 모두 정의하는 강력한 네트워크 보안을 제공."



kt cloud

FWaaS 는 OVN 에서 어떻게 구현 될까?

Neutron FWaaS → OVN Logical Router ACL 변환

Neutron Firewall Group Create

```
# openstack firewall group create --name <FWaaS Group Name>
```

```
+--------------------+--------------------------------------+
| Field              | Value                                |
+--------------------+--------------------------------------+
| Description        |                                      |
| Egress Policy ID   | None                                 |
| ID                 | a66ce4f0-4a80-4747-892f-a13260fd769c |
| Ingress Policy ID  | None                                 |
| Name               | test-fwaas                           |
| Ports              | []                                   |
| Project            | 9745b7b8dc9942e3856ce9d321ad0fd5     |
| Shared             | False                                |
| State              | UP                                   |
| Status             | INACTIVE                             |
| created_at         | 2025-08-21T10:11:11Z                 |
| revision_number    | 0                                    |
| tags               | []                                   |
| updated_at         | 2025-08-21T10:11:11Z                 |
+--------------------+--------------------------------------+
```

Firewall Group 을 생성하면 Default Rule 로 인해 Port Group, ACL 이 생성.

적용

적용

Port Group

```
# ovn-nbctl list port-group
```

```
_uuid               : 07bc9f72-ea2f-4367-a487-f7aab2a45831
acls                : [4a92a66d-ce8c-4edf-8a7b-47d65e351973, 6cf64843-2449-42a7-a247-2130f30d93be,
891ca3e4-b7a8-4583-a61f-9d2242a5eeef, 9a7813c2-efbc-48f9-a1ab-f197972e3fef]
external_ids        : {"neutron:firewall_group_id"="a66ce4f0-4a80-4747-892f-a13260fd769c"}
name                : pg_a66ce4f0_4a80_4747_892f_a13260fd769c
ports               : []
```

ACL Rule (4가지 기본 ACL 생성)

```
_uuid               : 4a92a66d-ce8c-4edf-8a7b-47d65e351973
action              : drop
direction           : to-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "outport == @pg_a66ce4f0_4a80_4747_892f_a13260fd769c && ip6"
priority            : 1001

_uuid               : 6cf64843-2449-42a7-a247-2130f30d93be
action              : drop
direction           : from-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "inport == @pg_a66ce4f0_4a80_4747_892f_a13260fd769c && ip4"
priority            : 1001

_uuid               : 891ca3e4-b7a8-4583-a61f-9d2242a5eeef
action              : drop
direction           : to-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "outport == @pg_a66ce4f0_4a80_4747_892f_a13260fd769c && ip4"
priority            : 1001

_uuid               : 9a7813c2-efbc-48f9-a1ab-f197972e3fef
action              : drop
direction           : from-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "inport == @pg_a66ce4f0_4a80_4747_892f_a13260fd769c && ip6"
priority            : 1001
```

FWaaS 의 Default Rules 과 Quota

FWaaS Group 생성 시 양 방향 drop 규칙이 Group 에 생성 됨.

Firewall Default ACL

```
_uuid               : 3cef3ac2-d7cb-4778-852c-6fff1d975c3f
action              : drop
direction           : to-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "outport == @pg_<pg_id> && ip4"
priority            : 1001

_uuid               : d539bf66-a181-4317-96e1-714514525d5c
action              : drop
direction           : to-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "outport == @pg_<pg_id> && ip6"
priority            : 1001

_uuid               : 168588ea-e75c-49ed-bc28-ffdd5389f7dd
action              : drop
direction           : from-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "inport == @pg_<pg_id> && ip6"
priority            : 1001

_uuid               : bc3a188d-c642-441c-97a8-883c46abf30d
action              : drop
direction           : from-lport
external_ids        : {"neutron:firewall_rule_id"=default_rule}
match               : "inport == @pg_<pg_id> && ip4"
priority            : 1001
```

FWaaS 는 우선 순위로 인해 Quota 제한을 가지게 됨.

Firewall Group quota 제한

Default quota

구분	기본 값
Firewall Group	10 (quota_firewall_group = 10)
Firewall policy	10 (quota_firewall_policy = 10)
Firewall Rule	100 (quota_firewall_rule = 100)

OVN ACL 우선 순위 (하드코드)

구분	기본 값
사용자 정의 Rule 시작점	2000
사용자 정의 Rule 정의	1999 - 1002
기본 DENY Rules	1001

FWaaS 규칙의 실제 OpenFlow 변환

OpenFlow 최종 규칙

```
 table=4 (ls_in_pre_acl      ), priority=3000 , match=(inport ==
@pg__<Firewall group ID> && ip4 && ip4.src == 203.0.113.100 && tc
p && tcp.dst == 80), action=(reg0[16] = 1; next;)
```

예약된 시스템 우선 순위와의 충돌을 방지하기 위해 ACL OFFSET 1000을 추가하여 동작.

*2000 + **1000** = 3000*
ACL + OFFSET = Openflow

OVN ACL 변환

```
_uuid               : 74991c01-fbbf-4fe5-9a0a-63300c9f9ab2
action              : allow-stateless
direction           : from-lport
external_ids        : {"neutron:firewall_rule_id"="<Firewall rule ID>"}
label               : 0
log                 : false
match               : "inport == @pg_<Firewall group ID> && ip4 && ip4.s
rc == 203.0.113.100 && tcp && tcp.dst == 80"
meter               : []
name                : []
options             : {}
priority            : 2000
severity            : []
tier                : 0
```

*FWaaS 의 경우 우선순위 반영을 위해 Rule, Policy 생성 후 **firewall group 설정**이 필요.*

Openstack Firewall Group Rule 생성

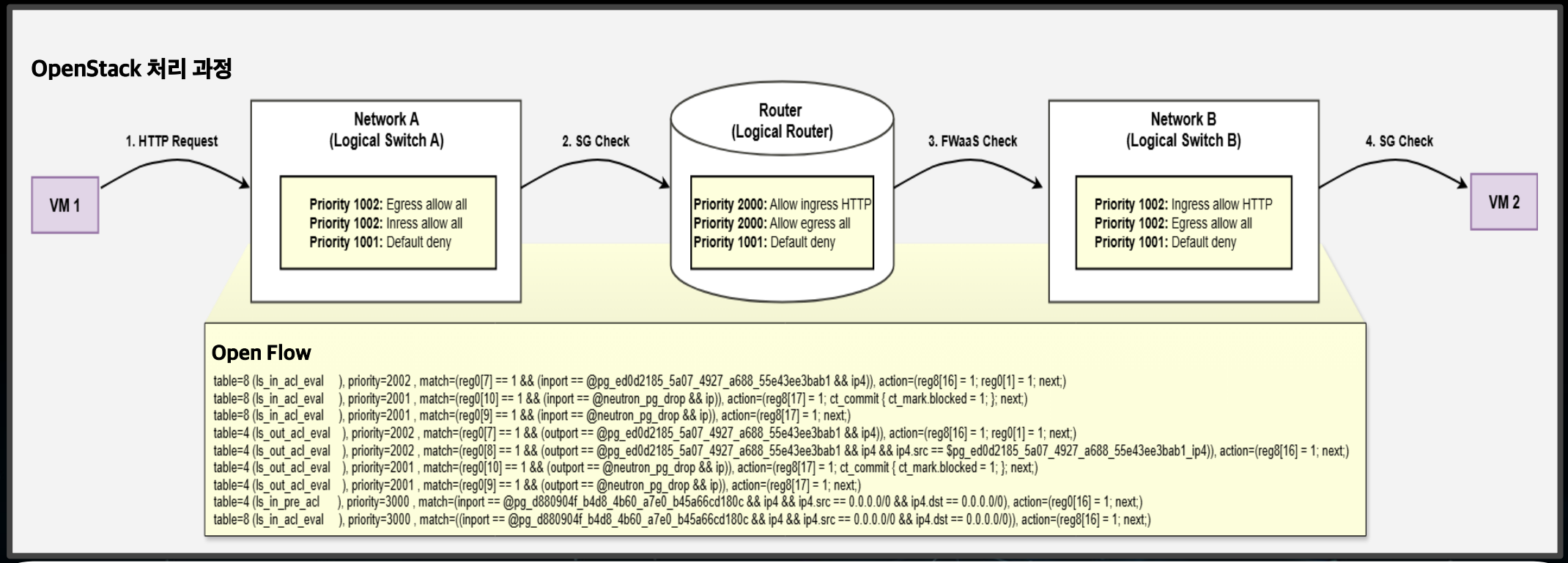
```
# openstack firewall group rule create --name deny-all-egress
--action deny --source-ip-address 0.0.0.0/0 --destination-ip-a
ddress 0.0.0.0/0
```

Openstack Firewall Group Policy 생성

```
# openstack firewall group policy create --firewall-rule 6c3f2
424-94e0-4f5f-8baf-7a432921d357 web-sg-policy-80
```

*FWaaS 의 경우 우선순위 **2000**부터 시작하며, firewall group 당 **최고 우선 순위도 2000** 임.*

Security Group과 FWaaS 의 전체 Flow



규칙 적용 순서:
Security Group: Logical Switch 레벨 (Port 별, Stateful)
FWaaS: Logical Router 레벨 (Port Group, Stateless)
처리 경로: Network(Security Group) -> Router(FWaaS) -> Network(Security Group)

우선 순위 처리:
높은 Priority 값부터 매칭
첫 번째 매칭되는 규칙 적용
Security Group: 1002(사용자 규칙), 1001(기본)
FWaaS: 2000(사용자 규칙), 1001(기본)

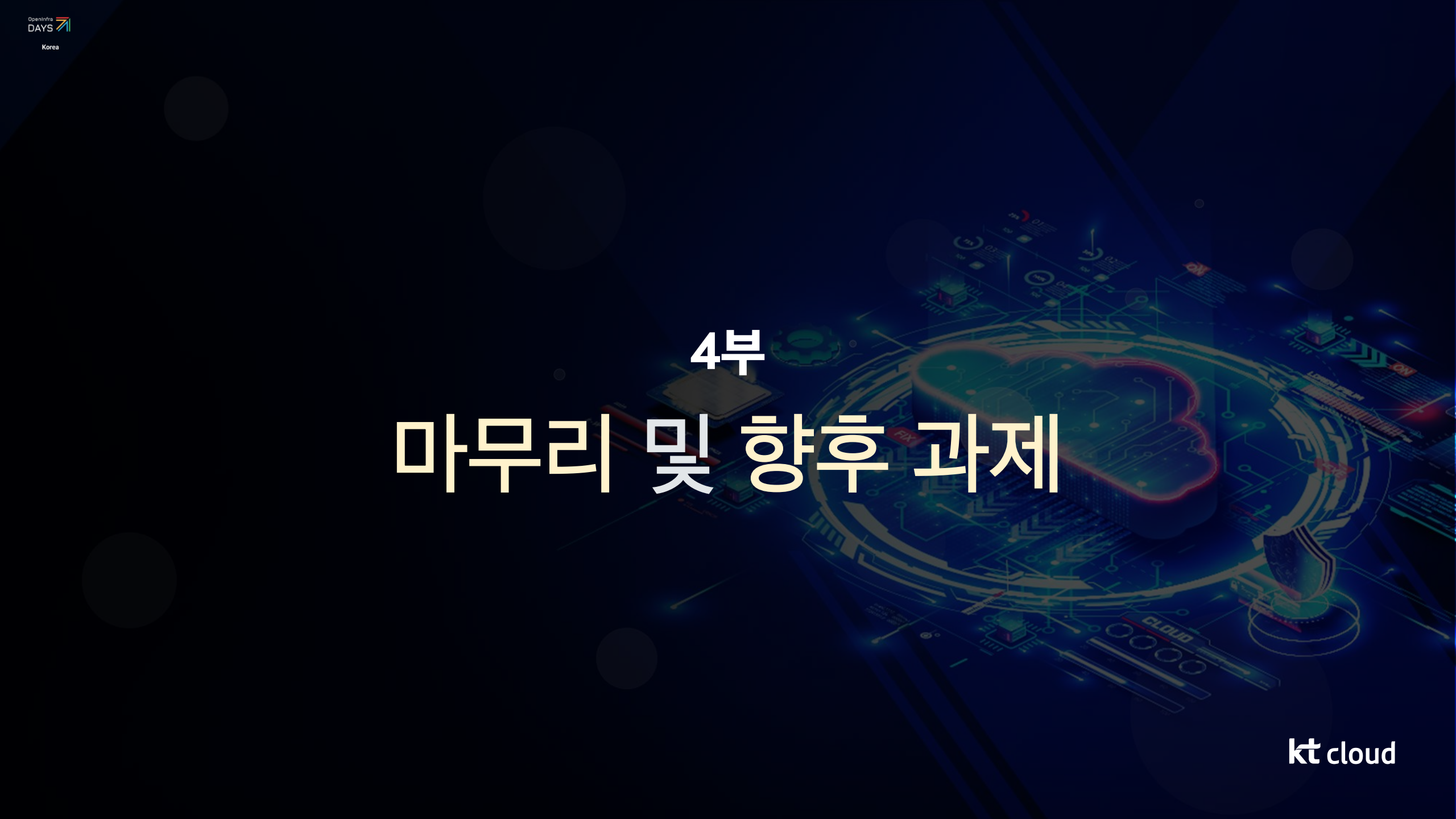
Security Group과 FWaaS 의 핵심 요약

Security Group vs FWaaS 구현 비교

구분	Security Group	FWaaS
관점	“Who”	“Where”
목적	VM 단위 상태 기반 방화벽	네트워크 경계
적용 지점	VM 가상 포트 (vNIC)	논리 라우터 포트
적용 범위	클라우드 인스턴스 단위	네트워크 경계 (vRouter)
구현 방식	OVN ACL 및 Port Group 을 통한 VM 별 규칙 적용	OVN ACL 및 Port Group 을 통한 라우터 별 규칙 적용
관리 방식	개별 인스턴스 관리	중앙 집중식 정책 관리
상태 관리	Stateful (allow-related)	Stateless (allow-stateless)
제한 사항	VM 중심으로 제한적	SG보다 관리 복잡성 높음, 많은 규칙 적용 시 라우터 성능 영향

“두 서비스는 상호 보완적으로 클라우드 환경에 다층적이고 견고한 보안 체계 구축”

OpenInfra DAYS Korea



4부

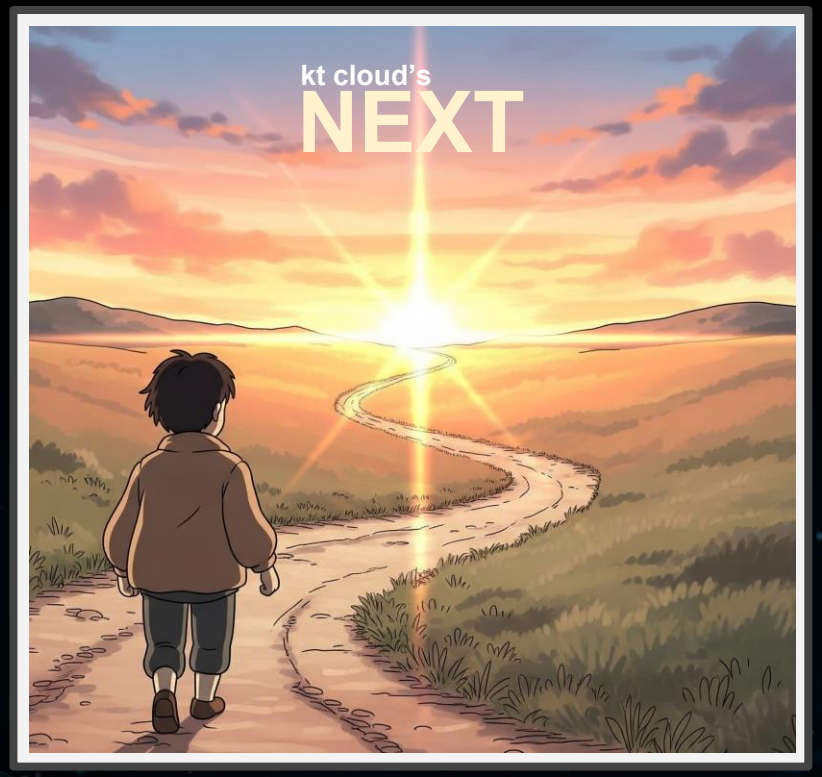
마무리 및 향후 과제

kt cloud

OpenInfra DAYS Korea

한 걸음씩, 앞으로!!

완벽하지 않지만, 계속 나아가며!!



성능 및 기능 관련 이슈

- 성능 저하: 대규모 환경에서 규칙 증가로 인한 처리 속도 저하 가능성
- Connectrion Tracking 이슈: Stateful특성으로 인한 규칙 우회 동작
- 규칙 우선순위 복잡성: Security Group와 FWaaS 간 우선순위 처리로 인한 예상치 못한 동작 가능성
- Flow 테이블 복잡성: 대량의 ACL 규칙으로 인한 OpenFlow 테이블에서 복잡성

운영 및 관리 관련 이슈

- 운영 부담: Security Group과 FWaaS의 다른 동작 레벨로 인한 디버깅과 트러블슈팅
- 가시성 부족: Neutron API에서 OVN ACL로의 변환 과정에서 실제 적용 상태 파악 어려움

결론

현재 여러 한계가 있고, 완벽하지는 않지만 한 걸음씩 나아가며 더욱 안전하고 효율적인 클라우드 환경을 만들어가고 있습니다.
한 걸음씩, 앞으로!!

kt cloud

OpenInfra DAYS Korea

감사합니다.

kt cloud