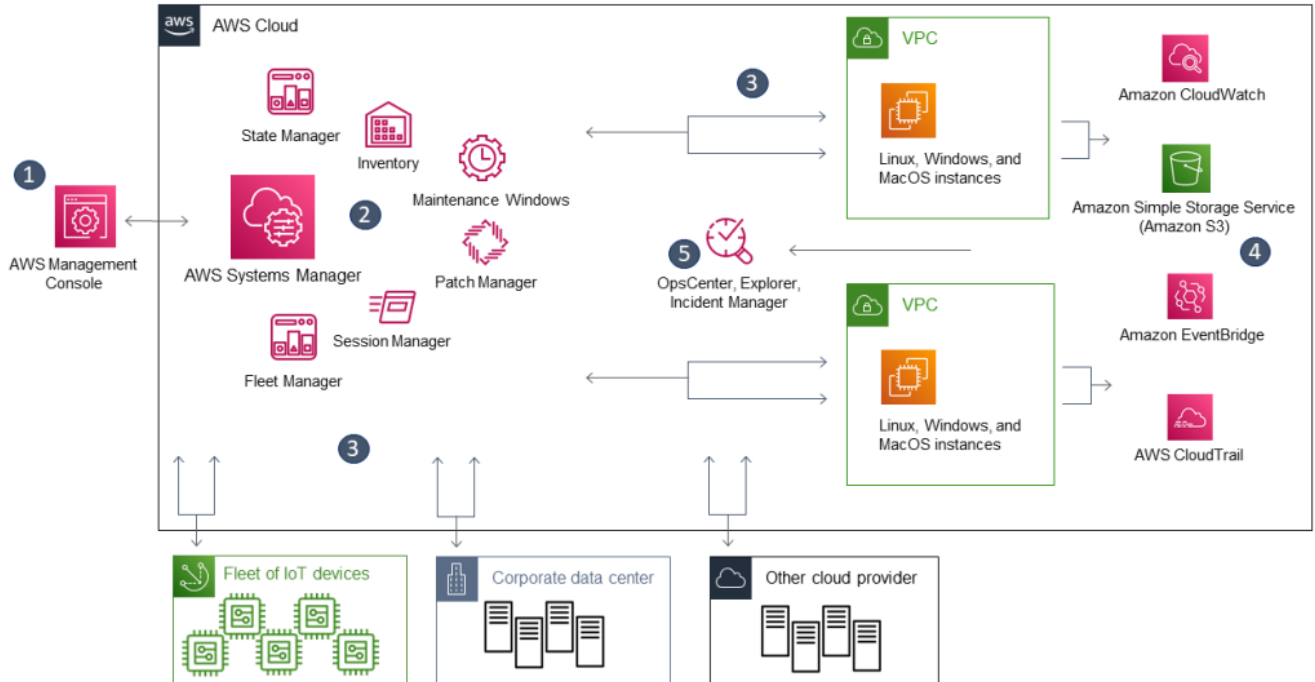


AWS SSM 을 통한 VPC Private Network 접근



목적

- VPC 내의 Private Network 에 외부 공인 대역에서 Shell 로 접근 (VPN 역할)

특징

- SSM 을 통해 AWS 연결을 하는 것으로, IGW 나 EIP 등의 VPC내 별도 제약이 필요없다
- EC2 SSH 접근 용도의 Password 나 Key-Pair 가 필요없다.
- Shell 전환으로 SSH 와 거의 동일하게 사용할 수 있다.
- AWS Client VPN 과 비교 시, 비용적으로 유리하다.

요구사항

- AWS CLi 를 설치 및 AWS 리소스에 접근할 수 있는 공인망 환경 (VM / CT / Server)
- 일부 리전의 경우 AWS Console 내에서 Cloudshell 사용 가능

설치 순서

- 1) Private Network 에 접근할 수 있는 IAM 권한 설정

- 2) EC2 에 IAM Role 추가
- 3) EC2 에 SSM Agent 설치
- 4) AWS CLI 를 통한 EC2 연결

1. Key 방식 전용의 IAM 생성 후 권한 설정

Add user

1 2 3

Set user details

You can add multiple users at once with the same access type and permissions. [Learn more](#)

User name*

[+ Add another user](#)

Select AWS access type

Select how these users will primarily access AWS. If you choose only programmatic access, it does NOT prevent users from accessing the an assumed role. Access keys and autogenerated passwords are provided in the last step. [Learn more](#)

- Select AWS credential type* ☒ **Access key - Programmatic access**
 Enables an **access key ID** and **secret access key** for the AWS API, CLI, SDK, and other development tools.
- ☐ **Password - AWS Management Console access**
 Enables a **password** that allows users to sign-in to the AWS Management Console.

Add user

1 2 3 4 5

 **Success**
 You successfully created the users shown below. You can view and download user security credentials. You can also email users instructions for signing in to the AWS Management Console. This is the last time these credentials will be available to download. However, you can create new credentials at any time.

Users with AWS Management Console access can sign-in at: <https://hostway-bmt.signin.aws.amazon.com/console>

 Download .csv

	User	Access key ID	Secret access key
▶	✓ SSM-Only	AKIAQ25632EPN7T2FFVT 	***** Show

- 관리형 정책을 만들어서 IAM 에 할당한다.

arn 은 리전과 어카운트와 접근할 EC2 ID 를 적는다

```
{
  "Version": "2012-10-17",
  "Statement": [
```

```

{
  "Effect": "Allow",
  "Action": [
    "ssm:StartSession"
  ],
  "Resource": [
    "arn:aws:ec2:us-west-2:1234567890:instance/i-
ahe52134fxed6"
  ]
},
{
  "Effect": "Allow",
  "Action": [
    "ssm:TerminateSession"
  ],
  "Resource": [
    "arn:aws:ssm:*:*:session/${aws:username} - *"
  ]
}
]
}

```

Create policy

Review policy

Before you create this policy, provide the required information and review this policy.

Name*

Maximum 128 characters. Use alphanumeric and '+', '@', '-' characters.

Summary

Filter			
Service ▾	Access level	Resource	Request condition
Allow (1 of 321 services) Show remaining 320			
Systems Manager	Limited: Write	Multiple	None

2. IAM Custom Role 을 생성한 후 VPC 내의 EC2 에 설정한 다

Step 1

Select trusted entity

Step 2

Add permissions

Step 3

Name, review, and create

Select trusted entity

Trusted entity type

☒ AWS service

Allow AWS services like EC2, Lambda, or others to perform actions in this account.

☐ AWS account

Allow entities in other AWS accounts belonging to you or a 3rd party to perform actions in this account.

☐ Web identity

Allows users federated by the specified external web identity provider to assume this role to perform actions in this account.

☐ SAML 2.0 federation

Allow users federated with SAML 2.0 from a corporate directory to perform actions in this account.

☐ Custom trust policy

Create a custom trust policy to enable others to perform actions in this account.

Use case

Allow an AWS service like EC2, Lambda, or others to perform actions in this account.

Common use cases

☒ EC2

Allows EC2 instances to call AWS services on your behalf.

☐ Lambda

Allows Lambda functions to call AWS services on your behalf.

Use cases for other AWS services:

Choose a service to view use case

- Role 은 SSM InstanceCore 만 선택

Step 1

Select trusted entity

Step 2

Add permissions

Step 3

Name, review, and create

Add permissions

Permissions policies (Selected 1/754)

Choose one or more policies to attach to your new role.

Filter policies by property or policy name and press enter

14 matches

"SSM" X

Clear filters

☐	Policy name ↗	Type	Description
☐	AmazonEC2RoleforSSM	AWS m...	This policy will soon be deprecated. Please use AmazonSSManagedInstanceCore policy to enable AWS Systems Manager servic...
☐	AmazonSSMAutomationApproverAccess	AWS m...	Provides access to view automation executions and send approval decisions to automation waiting for approval
☒	AmazonSSManagedInstanceCore	AWS m...	The policy for Amazon EC2 Role to enable AWS Systems Manager service core functionality
☐	AmazonSSMDirectoryServiceAccess	AWS m...	This policy allows SSM Agent to access Directory Service on behalf of the customer for domain-join the managed instance.
☐	AmazonSSMFullAccess	AWS m...	Provides full access to Amazon SSM.
☐	AmazonSSMAutomationRole	AWS m...	Provides permissions for EC2 Automation service to execute activities defined within Automation documents
☐	AmazonSSMReadOnlyAccess	AWS m...	Provides read only access to Amazon SSM.
☐	AmazonSSMMaintenanceWindowRole	AWS m...	Service Role to be used for EC2 Maintenance Window
☐	AWSResourceAccessManagerReadOnlyAccess	AWS m...	Provides read only access to AWS Resource Access Manager.

- Role Name 지정 후 생성

Step 1
Select trusted entity

Step 2
Add permissions

Step 3
Name, review, and create

Name, review, and create

Role details

Role name

Enter a meaningful name to identify this role.

Maximum 128 characters. Use alphanumeric and '+=, @-_' characters.

Description

Add a short explanation for this policy.

Allows EC2 instances to call AWS services on your behalf.

Maximum 1000 characters. Use alphanumeric and '+=, @-_' characters.

Step 1: Select trusted entities

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Action": [  
7         "sts:AssumeRole"  
8       ],  
9       "Principal": {  
10        "Service": [  
11          "ec2.amazonaws.com"  
12        ]  
13      }  
14    ]  
15  }
```

EC2 의 보안 설정에서 해당 롤을 추가한다

Modify IAM role [Info](#)

Attach an IAM role to your instance.

Instance ID

 i-064f7ebc0bed75c74 (BAEK-Bastion)

IAM role

Select an IAM role to attach to your instance or create a new role if you haven't created any. The role you select replaces any roles that are currently attached to your instance.



[Create new IAM role](#) 

Cancel

Save

3. EC2 에 SSM-Agent 를 설치한다

Aamazon 2 리눅스 인스턴스의 경우에는 기본 설정되어 있을 수 있다.

```
[root@ip-10-10-20-201 ~]# sudo yum install -y
https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux
_amd64/amazon-ssm-agent.rpm
```

Loaded plugins: extras_suggestions, langpacks, priorities, update-motd

```
Cannot open:
https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux
_amd64/amazon-ssm-agent.rpm. Skipping.
```

Error: Nothing to do

```
[root@ip-10-10-20-201 ~]# wget
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest
/linux_amd64/amazon-ssm-agent.rpm
```

--2022-03-29 02:49:06--

```
https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest
/linux_amd64/amazon-ssm-agent.rpm
```

Resolving s3.amazonaws.com (s3.amazonaws.com)... 52.217.196.240

Connecting to s3.amazonaws.com (s3.amazonaws.com)|52.217.196.240|:443... connected.

HTTP request sent, awaiting response... 200 OK

Length: 26724168 (25M) [binary/octet-stream]

Saving to: 'amazon-ssm-agent.rpm'

```
100%[=====
=====>] 26,724,168
12.7MB/s in 2.0s
```

2022-03-29 02:49:08 (12.7 MB/s) - 'amazon-ssm-agent.rpm' saved [26724168/26724168]

설치 후 자동시작 활성화한다.

```
[root@ip-10-10-20-201 ~]# rpm -Uvh amazon-ssm-agent.rpm
```

warning: amazon-ssm-agent.rpm: Header V4 RSA/SHA1 Signature, key ID 693eca21: NOKEY

Preparing...

[100%]

Updating / installing...

1:amazon-ssm-agent-3.1.1080.0-1

[100%]

Created symlink from /etc/systemd/system/multi-user.target.wants/amazon-ssm-agent.service to

```
/etc/systemd/system/amazon-ssm-agent.service.
```

```
[root@ip-10-10-20-201 ~]# systemctl enable amazon-ssm-agent
```

```
[root@ip-10-10-20-201 ~]# systemctl start amazon-ssm-agent
```

```
[root@ip-10-10-20-201 ~]# systemctl status amazon-ssm-agent
```

```
-- amazon-ssm-agent.service - amazon-ssm-agent
   Loaded: loaded (/etc/systemd/system/amazon-ssm-agent.service; enabled; vendor preset: enabled)
   Active: active (running) since Tue 2022-03-29 02:53:54 UTC; 21s ago
     Main PID: 3355 (amazon-ssm-agent)
       CGroup: /system.slice/amazon-ssm-agent.service
               └─3355 /usr/bin/amazon-ssm-agent
               └─3382 /usr/bin/ssm-agent-worker
```

```
Mar 29 02:53:54 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:54 INFO Agent will take identity f...EC2
```

```
Mar 29 02:53:54 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:54 INFO [amazon-ssm-agent] using n...IPC
```

```
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:54 INFO [amazon-ssm-agent] using n...IPC
```

```
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:54 INFO [amazon-ssm-agent] using n...IPC
```

```
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:54 INFO [amazon-ssm-agent] amazon-...0.0
```

```
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:54 INFO [amazon-ssm-agent] OS: lin...d64
```

```
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:54 INFO [CredentialRefresher] Iden...her
```

```
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal amazon-ssm-agent[3355]: 2022-03-29 02:53:55 INFO [amazon-ssm-agent] [LongRu...ess
```

```
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal
amazon-ssm-agent[3355]: 2022-03-29 02:53:55 INFO [amazon-ssm-
agent] [LongRu...ted
Mar 29 02:53:55 ip-10-10-20-201.us-west-2.compute.internal
amazon-ssm-agent[3355]: 2022-03-29 02:53:55 INFO [amazon-ssm-
agent] [LongRu...nds
Hint: Some lines were ellipsized, use -l to show in full.
```

4. AWS Cli 에서 SSM 을 통한 EC2 접속

외부 CT 에 AWScli 를 설치 후 IAM API Key 권한을 설정한다.

```
## 접속할 Client의 IP 확인
$ curl http://icanhazip.com
1.2.3.4
```

AWScli 설치

(중요) SSM의 Session-Plugin 을 사용하기 위해서는 AWScli 1.16 버전 이상 이어야 한다.

```
$ curl
"https://awscli.amazonaws.com/awscli-exe-linux-x86_64.zip" -o
"awscliv2.zip"
unzip awscliv2.zip
sudo ./aws/install
```

Linux 용 SSM Session-Manager Plugin 설치

```
$ curl
"https://s3.amazonaws.com/session-manager-downloads/plugin/latest/linux_64bit/session-manager-plugin.rpm" -o "session-
manager-plugin.rpm"
$ rpm -Uvh session-manager-plugin.rpm
$ session-manager-plugin
The Session Manager plugin was installed successfully. Use the
AWS CLI to start a session.
```

접근 정보 설정

본문 내용의 키는 임의 생성한 내용.

```
$ aws configure
AWS Access Key ID [None]: AKIAQ25632EPN7T7FFVT
AWS Secret Access Key [None]:
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
Default region name [None]: us-west-2
```


Default output format [None]: json

SSM 을 통한 실제 EC2 에 접근

```
[root@node1 ~]# aws ssm start-session --target i-064f7ebc0bed75c74
```

Starting session with SessionId: SSM-Only-0a9041d6b13f368ce

sh-4.2\$ **bash**

[ssm-user@ip-10-10-20-201 bin]\$ ifconfig

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 9001
inet 10.10.20.201 netmask 255.255.255.0 broadcast 10.10.20.255

inet6 fe80::aa:14ff:fed7:abd prefixlen 64 scopeid 0x20<link>

ether 02:aa:14:d7:0a:bd txqueuelen 1000 (Ethernet)

RX packets 31846 bytes 8338621 (7.9 MiB)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 29180 bytes 6068149 (5.7 MiB)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536

inet 127.0.0.1 netmask 255.0.0.0

inet6 ::1 prefixlen 128 scopeid 0x10<host>

loop txqueuelen 1000 (Local Loopback)

RX packets 0 bytes 0 (0.0 B)

RX errors 0 dropped 0 overruns 0 frame 0

TX packets 0 bytes 0 (0.0 B)

TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[ssm-user@ip-10-10-20-201 bin]\$

※ 참고자료

<https://docs.aws.amazon.com/cli/latest/userguide/getting-started-install.html>

https://docs.aws.amazon.com/ko_kr/systems-manager/latest/userguide/session-manager-working-with-install-plugin.html

https://docs.aws.amazon.com/ko_kr/systems-manager/latest/userguide/session-manager-getting-started.html