

Laboratorio Configuración System Manager.

Contenido

Laboratorio – Introducción System Manager	2
Objetivo Práctica Laboratorio	2
System Manager.....	2
Despliegue Infraestructura – Añadir Recursos.....	2
Crear Stack - CloudFormation.	2
Instalación Agente System Manager.....	4
Roles de Administración.....	5
Configuración System Manager	10
Etiquetas.....	10
Grupo de Recursos.	13
Información integrada.....	15
Inventario.	16
Session Manager.	21
State Manager.....	24
Start/Stop Instance.	28
Bibliografía	34

Laboratorio – Introducción System Manager

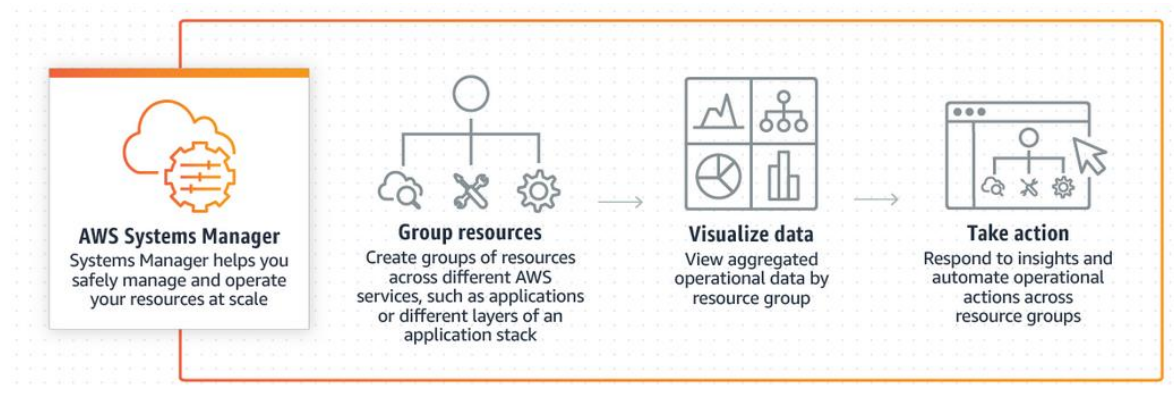
Objetivo Práctica Laboratorio

IMPLEMENTAR la herramienta AWS System Manager para gestionar de manera centralizada los recursos de AWS.

System Manager

AWS Systems Manager brinda un panel de control centralizado de la infraestructura en AWS. A través del System Manager puede ver los datos operativos de varios servicios de AWS y le permite automatizar tareas operativas en todos sus recursos de AWS. Con Systems Manager puede agrupar los recursos de cada aplicación, por ejemplo, instancias de Amazon EC2, buckets de Amazon S3 o instancias de Amazon RDS. Adicionalmente, AWS System Manager permite ver datos operativos para monitoreo y solución de problemas y actuar sobre los grupos de recursos.

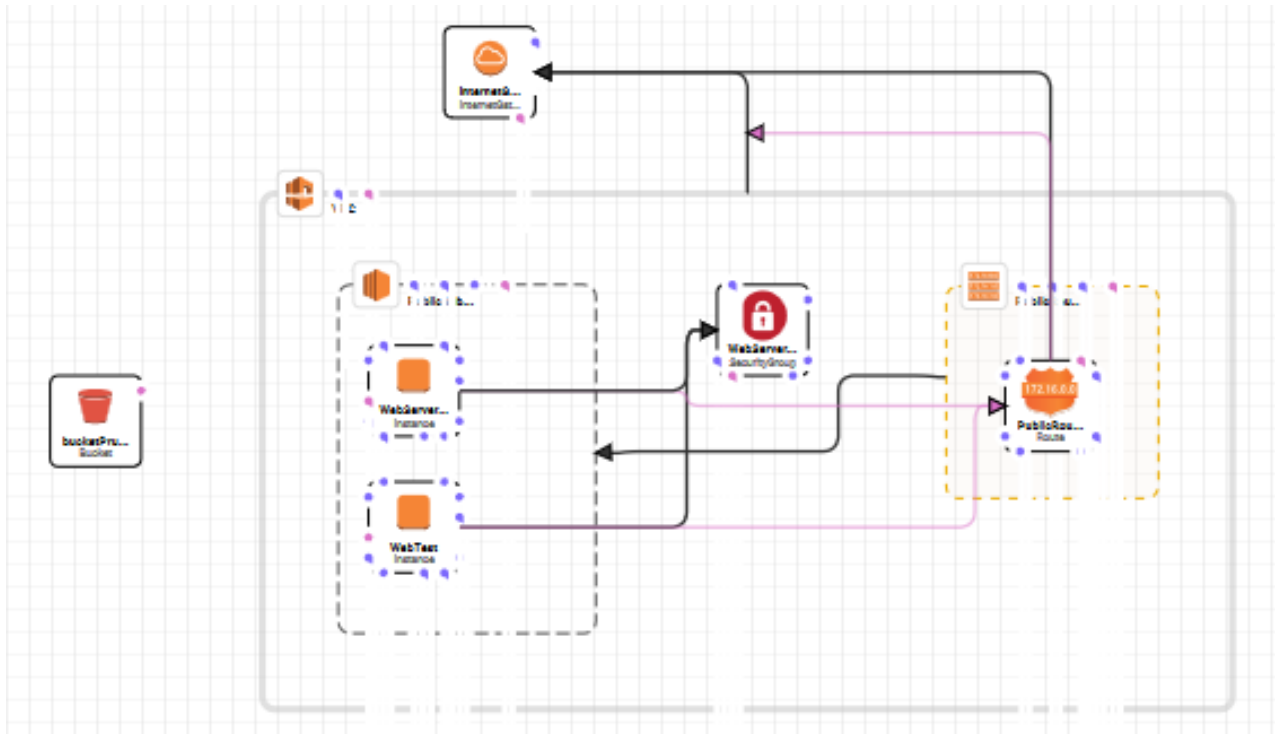
En resumen, Systems Manager simplifica la administración de aplicaciones y recursos, agiliza la detección y resolución de problemas operativos, y facilita el uso y la administración de la infraestructura de manera segura a escala. El diagrama 1, describe el funcionamiento general de AWS System Manager.



Despliegue Infraestructura – Añadir Recursos.

Crear Stack - CloudFormation.

Para agilizar el despliegue de la infraestructura en AWS, vamos a utilizar la herramienta AWS CloudFormation para ejecutar una plantilla de infraestructura como código y crear una pila de recursos. La infraestructura que vamos a desplegar contiene 2 instancias EC2, una de producción y otra de pruebas. Además, tenemos un bucket S3 para almacenamiento de archivos. El siguiente es el diagrama final de la infraestructura a desplegar:



Para este laboratorio vamos a utilizar RedHat como sistema operativo de las instancias. Para esto debemos cambiar los mappings de las AMIs en nuestra región.

```
"us-east-1": {
  "HVM64": "ami-098bb5d92c8886ca1",
  "HVMG2": "ami-07bd28c96286169fa"
},
```

Es importante considerar que, para este laboratorio, la instancia de prueba debe mostrar solamente un HTML con una etiqueta H1 que diga su nombre. Para la instancia de producción, deberá cargar una aplicación web previamente construida. Para instalarla debe considerar los siguientes comandos

```
wget https://us-west-2-tcprod.s3.amazonaws.com/courses/AWS-100-
CCA/v3.1.21/lab8-ha/scripts/phpapp.zip

unzip phpapp.zip -d /var/www/html/
```

La aplicación debe quedar instalada y funcionando desde la ejecución de la plantilla.

Adicionalmente, antes de ejecutar la plantilla, vamos a crear una nueva llave para nuestros servidores en RedHat. Este par de llaves serán las que utilizaremos para lograr la conexión con el servidor Red Hat e instalar los componentes necesarios.

Create Key Pair

Import Key Pair

Delete

Filter by attributes or search by keyword

<input type="checkbox"/>	Key pair name	Fingerprint
<input type="checkbox"/>	clave	58:18:74:44:3f:53:b7:92:77:33:66:c7:e9:ff:da:f5:86:12:8f:9b
<input type="checkbox"/>	LaboratoriollS	2f:b4:ae:48:42:b1:a9:63:9c:df:45:b5:e0:1c:47:11:ff:75:f9:d6
<input type="checkbox"/>	Pacifi2	da:d0:f2:11:ba:78:be:c6:75:4b:b8:bb:4e:b9:13:07:66:76:89:a9
<input type="checkbox"/>	Pacifi2_2	b2:ef:75:3a:9f:8f:ca:a9:ff:d9:fe:70:22:7b:5d:15:e6:c6:19:c8
<input type="checkbox"/>	pacifi3	53:fd:eb:cc:a6:ef:ef:50:30:06:12:ae:a7:63:2c:e3:c0:8a:ea:9e
<input type="checkbox"/>	pacifi5	bb:a2:95:2c:75:5e:b2:09:d9:bb:c7:4c:a4:cf:a7:2d:27:e5:77:8d
<input type="checkbox"/>	Pacifi6	db:a8:75:83:4f:92:2f:9e:07:27:e3:33:b2:d0:70:8b:78:ab:58:5e
<input type="checkbox"/>	pacifi7	fb:9b:a4:81:41:aa:22:8e:ac:68:68:31:24:18:c6:b6:e7:61:e7:75
<input type="checkbox"/>	pacifi8	85:c6:7e:87:c0:10:78:e7:5f:9d:42:f4:14:5a:60:2a:ac:15:d4:31
<input type="checkbox"/>	Pacifi8_1	bd:31:cc:4a:9b:14:02:8f:8c:17:d7:6a:df:11:2c:7a:88:1c:af:f4

Select a key pair

Descargamos el archivo .pem y lo dejamos guardado para el futuro.

Instalación Agente System Manager.

Para gestionar las instancias de EC2, es necesario realizar la instalación del agente del System Manager. Por defecto en las instancias de Amazon Linux se encuentra instalado el agente del SSM. En caso que usted esté utilizando un sistema operativo diferente, cómo puede seguir la siguiente guías de instalación: https://docs.aws.amazon.com/es_es/systems-manager/latest/userguide/ssm-agent.html

Para el desarrollo de este laboratorio, debemos conectarnos a nuestras instancias EC2 creadas con la plantilla siguiendo la información del siguiente enlace: https://docs.aws.amazon.com/es_es/AWSEC2/latest/UserGuide/putty.html.

Una vez conectados, ingresamos con el usuario ec2-user y ejecutamos los siguientes comandos:

```
sudo yum install -y https://s3.amazonaws.com/ec2-downloads-windows/SSMAgent/latest/linux amd64/amazon-ssm-agent.rpm
sudo status amazon-ssm-agent
```

Una vez ejecutados ambos comandos, deberá obtener el siguiente resultado:

```

Dependencies Resolved

=====
Package                Arch          Version        Repository      Size
=====
Updating:
amazon-ssm-agent        x86_64        2.3.634.0-1    /amazon-ssm-agent 61 M

Transaction Summary
=====
Upgrade 1 Package

Total size: 61 M
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
amazon-ssm-agent stop/waiting
  Updating   : amazon-ssm-agent-2.3.634.0-1.x86_64                1/2
  Cleanup    : amazon-ssm-agent-2.2.916.0-1.amzn1.x86_64         2/2
amazon-ssm-agent start/running, process 3106
  Verifying  : amazon-ssm-agent-2.3.634.0-1.x86_64                1/2
  Verifying  : amazon-ssm-agent-2.2.916.0-1.amzn1.x86_64         2/2

Updated:
  amazon-ssm-agent.x86_64 0:2.3.634.0-1

Complete!

```

```

[ec2-user@ip-10-0-0-238 ~]$ sudo status amazon-ssm-agent
amazon-ssm-agent start/running, process 3106

```

Con la instalación del agente, el servicio de System Manager podrá ejecutar comando, servicios, cambiar configuraciones y demás acciones en la instancia EC2.

Roles de Administración.

Un elemento importante para lograr que el System Manager administre nuestros recursos de AWS, es la creación de 2 roles que den permiso al servicio del System Manager en modificar y acceder a los recursos que se van a gestionar de manera centralizada. Para este laboratorio vamos a crear un Rol de Gestión de Instancias EC2 y un rol de Automatización.

Rol Instancia EC2.

Recordemos que los roles, nos permiten asignar políticas a los recursos de AWS, por lo que para su configuración nos dirigimos al servicio IAM, sección de Role y damos clic en crear un nuevo Rol:

Panel

Grupos

Usuarios

Roles

Políticas

Proveedores de identidad

Configuración de cuenta

Informe de credenciales

Claves de cifrado

Crear un rol

Eliminar el rol

Buscar


Mostrando 23 resultados

Nombre de rol	Descripción	Entidades de confianza
<input type="checkbox"/> AWSServiceRoleForConfig		Servicio de AWS: config (Rol vinculado a servi...
<input type="checkbox"/> AWSServiceRoleForElasticLoad...	Allows ELB to call AWS services on your behalf.	Servicio de AWS: elasticloadbalancing (Rol vin...
<input type="checkbox"/> AWSServiceRoleForOrganizations	Service-linked role used by AWS Organizations to enable integratio...	Servicio de AWS: organizations (Rol vinculado ...
<input type="checkbox"/> AWSServiceRoleForSupport	Enables resource access for AWS to provide billing, administrative ...	Servicio de AWS: support (Rol vinculado a ser...
<input type="checkbox"/> AWSServiceRoleForTrustedAdvi...	Access for the AWS Trusted Advisor Service to help reduce cost, in...	Servicio de AWS: trustedadvisor (Rol vincula...
<input type="checkbox"/> DanielNinoRole		Servicio de AWS: lambda
<input type="checkbox"/> DanielRole		Servicio de AWS: lambda
<input type="checkbox"/> Edireoknition_1	Allows Lambda functions to call AWS services on your behalf.	Servicio de AWS: lambda
<input type="checkbox"/> IsPalindromeRole		Servicio de AWS: lambda


Elegimos el servicio de EC2, para el rol. Esto se aprecia en la siguiente captura:

Crear un rol

Seleccionar el tipo de entidad de confianza



Servicio de AWS
EC2, Lambda y otros



Otra cuenta de AWS
Perteneciente a usted o a un tercero

Permite a los servicios de AWS realizar acciones en su nombre. [Más información](#)

Elegir el servicio que utilizará este rol

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

[API Gateway](#)

[CodeDeploy](#)

[EKS](#)

[AWS Backup](#)

[Comprehend](#)

[EMR](#)

[AWS Support](#)

[Config](#)

[ElastiCache](#)

[Amplify](#)

[Connect](#)

[Elastic Beanstalk](#)


En la política, asociamos la política de AWS llamada **AmazonEC2RoleforSSM**.

Crear un rol

▼ Attach políticas de permisos

Elija una o varias políticas para asociarlas al nuevo rol.

Crear una política

Filtrar políticas ▼		Q AmazonEC2RoleforSSM	
	Nombre de la política ▼	Utilizado como	
<input checked="" type="checkbox"/>	▶  AmazonEC2RoleforSSM	Ninguna	

Esta política cuenta con los permisos necesarios para la lograr que system manager pueda verificar nuestros recursos de AWS. Los siguientes son los permisos de la política:

Filtrar políticas ▼		Q AmazonEC2RoleforSSM	
	Nombre de la política ▼	Utilizado como	
	Systems Manager	Limitado: Enumeración, Lectura, Escritura	Todos los recursos
	SSM Messages	Acceso completo	Todos los recursos
	S3	Limitado: Enumeración, Lectura, Escritura	Todos los recursos
	EC2 Messages	Acceso completo	Todos los recursos
	EC2	Limitado: Enumeración	Todos los recursos
	Directory Service	Limitado: Enumeración, Escritura	Todos los recursos
	CloudWatch Logs	Limitado: Enumeración, Escritura	Todos los recursos
	CloudWatch	Limitado: Escritura	Todos los recursos

Damos Clic en Siguiente para configurar las etiquetas, en este punto podemos dejar las etiquetas en blanco. Por último, asignamos un nombre al rol y damos clic en crear rol.

Revisar

Proporcione la información requerida a continuación y revise este rol antes de crearlo.

Nombre de rol*

Utilice caracteres alfanuméricos y "+,=,_,@,-". 64 caracteres como máximo.

Descripción del rol

1000 caracteres como máximo. Utilice caracteres alfanuméricos y "+,=,_,@,-".

Entidades de confianza Servicio de AWS: ec2.amazonaws.com

Políticas  [AmazonEC2RoleforSSM](#)

Límite de permisos No se ha establecido un límite de permisos

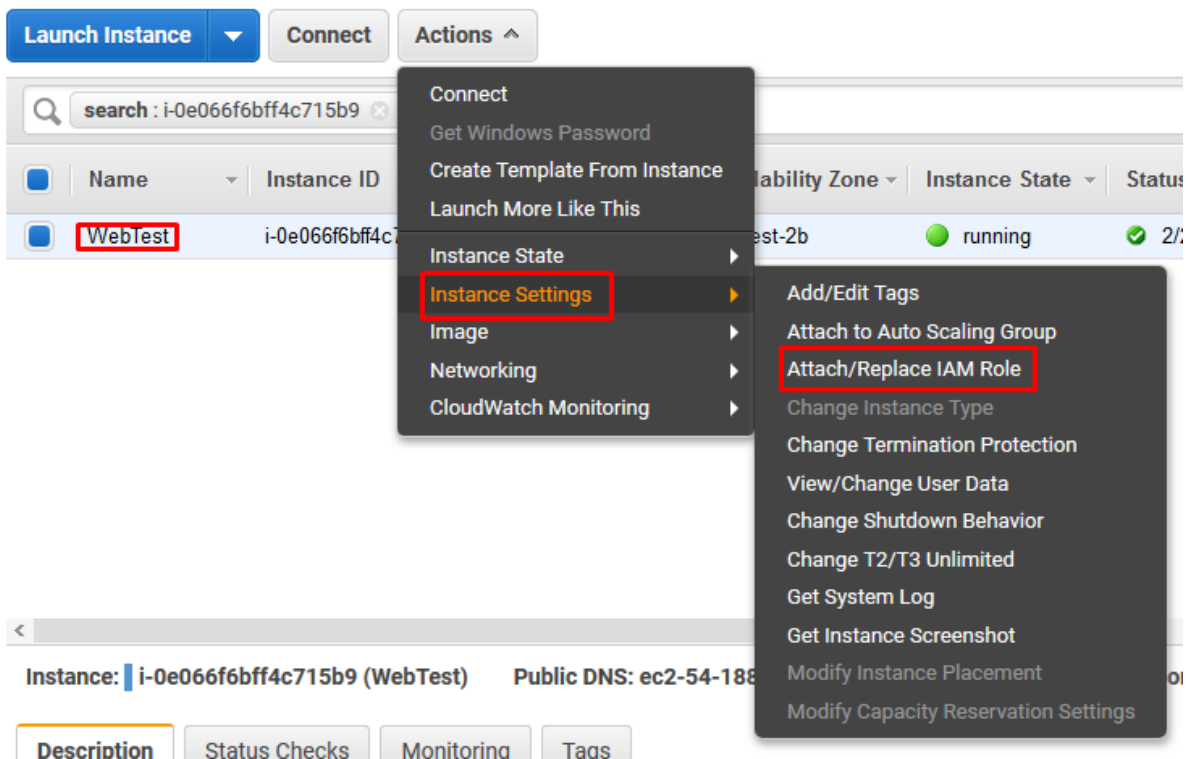
* Obligatorio

Cancelar

Anterior

Crear un rol

Ahora para terminar, debemos asignar el rol a cada una de nuestras instancias EC2. Lo hacemos de la siguiente manera:



The screenshot shows the AWS Management Console interface. At the top, there are buttons for 'Launch Instance', 'Connect', and 'Actions'. Below these is a search bar with the text 'search : i-0e066f6bffa4c715b9'. A table lists EC2 instances, with one instance named 'WebTest' (ID: i-0e066f6bffa4c715b9) highlighted. The 'Actions' menu is open, showing options like 'Connect', 'Create Template From Instance', and 'Launch More Like This'. The 'Instance State' menu is also open, showing options like 'Instance Settings', 'Image', 'Networking', and 'CloudWatch Monitoring'. The 'Instance Settings' sub-menu is open, showing options like 'Add/Edit Tags', 'Attach to Auto Scaling Group', and 'Attach/Replace IAM Role'. The 'Attach/Replace IAM Role' option is highlighted. Below the table, there is a section for the selected instance 'i-0e066f6bffa4c715b9 (WebTest)', showing its 'Public DNS' as 'ec2-54-188-100-100.us-east-2b' and tabs for 'Description', 'Status Checks', 'Monitoring', and 'Tags'.

Attach/Replace IAM Role

Select an IAM role to attach to your instance. If you don't have any IAM roles, choose Create new IAM role to create a role in the IAM console. If an IAM role is already attached to your instance, the IAM role you choose will replace the existing role.

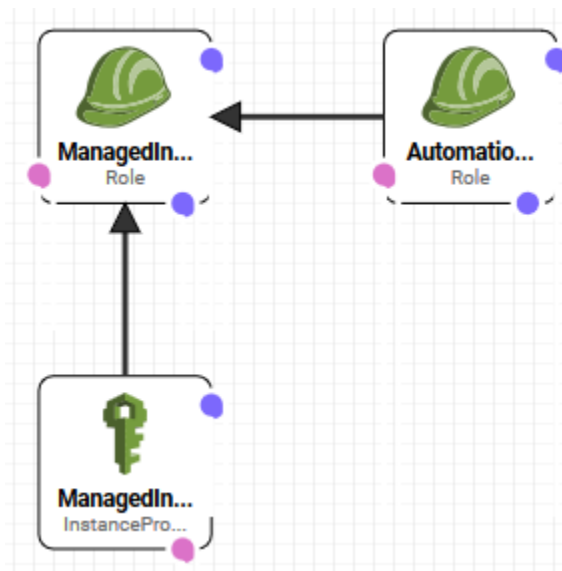
Instance ID i-04754d7c6dbdae9a3 () ⓘ

IAM role* EC2RoleForSSM-Pacifi ▼ ⌂ Create new IAM role ⓘ

Rol de Automatización.

Este segundo Rol, es el que dará permisos a los servicios de automatización del System Manager en modificar nuestros recursos de AWS. El primer Rol que creamos nos ayuda a monitorear y este rol a cambiar configuración y realizar labores de monitoreo. Debido a que las rutinas de automatización se ejecutan siempre que se cumplan unas condiciones sin necesidad de aprobación del usuario, es importante dar solamente los permisos necesarios a cada rutina para evitar generar errores. Además, al utilizar las pilas de recursos, en caso de un error es posible borrar los roles y quitar los permisos a las rutinas automáticas y detener la ejecución.

Para agilizar esta tarea, el rol ya fue creado en la cuenta de la universidad utilizando la siguiente plantilla de CloudFormation <http://bit.ly/2HL86tp>. El siguiente diagrama resume la operación de la plantilla:



Una vez ejecutada la plantilla deberá obtener las siguientes salidas y ver los siguientes recursos:

AutomatizacionRole

Stack info

Events

Resources

Outputs

Parameters

Template

Actions

Overview

Stack name

AutomatizacionRole

Stack ID

arn:aws:cloudformation:us-west-2:682086073548:stack/AutomatizacionRole/84228560-556a-11e9-80f8-0650fec6e554

Root stack

-

Stack status

CREATE_COMPLETE

Resources (3)

Q

Search resources

Logical ID	Physical ID	Type	Status
AutomationServiceRole	AutomationServiceRole	AWS::IAM::Role	CREATE_COMPLETE
ManagedInstanceProfile	ManagedInstanceProfile	AWS::IAM::InstanceProfile	CREATE_COMPLETE
ManagedInstanceRole	AutomatizacionRole-ManagedInstanceRole-1AM4E0NLHRJEA	AWS::IAM::Role	CREATE_COMPLETE

Configuración System Manager

Etiquetas.

Para crear grupos de recursos, se recomienda utilizar las etiquetas en AWS. En este caso vamos a crear una etiqueta por cada uno de los recursos creados con nuestra plantilla llamada ambiente. Ambiente será la llave de la etiqueta y la descripción será: producción o pruebas.

Con las etiquetas podremos agrupar recursos de AWS según sus características o funcionalidades dentro de nuestra infraestructura.

Las siguientes capturas muestra la configuración de etiqueta:

WebServer i-02e870bd8875a7dd6 t2.micro us-west-2b running 2/2 checks ... None ec2-34-220-101-197.us-... 34%

WebTest i-0e066fbff4c715b9 t2.micro us-west-2b running 2/2 checks ... None ec2-54-188-9-90.us-we-... 34%

Instance: i-02e870bd8875a7dd6 (WebServer) Public DNS: ec2-34-220-101-197.us-west-2.compute.amazonaws.com

Description Status Checks Monitoring **Tags**

Add/Edit Tags

Key	Value	
Name	WebServer	Hide Column
aws:cloudformation:logical-id	WebServerInstance	Show Column
aws:cloudformation:stack-id	arn:aws:cloudformation:us-west-2:682086073548:stack/PruebaSSM/e2925c70-5578-11e9-9b29-066b98e74c72	Show Column
aws:cloudformation:stack-name	PruebaSSM	Show Column

Add/Edit Tags



Apply tags to your resources to help organize and identify them.

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. [Learn more](#) about tagging your Amazon EC2 resources.

Key	Value	
<input type="text" value="Name"/>	<input type="text" value="WebServer"/>	ⓧ Hide Column
<input type="text" value="aws:cloudformation:logical-id"/>	<input type="text" value="WebServerInstance"/>	ⓧ Show Column
<input type="text" value="aws:cloudformation:stack-id"/>	<input type="text" value="arn:aws:cloudformation:us-"/>	ⓧ Show Column
<input type="text" value="aws:cloudformation:stack-name"/>	<input type="text" value="PruebaSSM"/>	ⓧ Show Column

Create Tag

Cancel

Save

Add/Edit Tags



Apply tags to your resources to help organize and identify them.

A tag consists of a case-sensitive key-value pair. For example, you could define a tag with key = Name and value = Webserver. [Learn more](#) about tagging your Amazon EC2 resources.

Key	Value	
Name	WebServer	Hide Column
aws:cloudformation:logical-id	WebServerInstance	Show Column
aws:cloudformation:stack-id	arn:aws:cloudformation:us-	Show Column
aws:cloudformation:stack-name	PruebaSSM	Show Column
Ambiente	Produccion	

Create Tag Cancel Save

WebServer	i-02e870bd8875a7dd6	t2.micro	us-west-2b	running	2/2 checks ...	None	ec2-34-220-101-197.us-...	34/2
WebTest	i-0e066f6bf4c715b9	t2.micro	us-west-2b	running	2/2 checks ...	None	ec2-54-188-9-90.us-we...	54/2

Instance: i-02e870bd8875a7dd6 (WebServer) Public DNS: ec2-34-220-101-197.us-west-2.compute.amazonaws.com

Description Status Checks Monitoring Tags

Add/Edit Tags

Key	Value	
Ambiente	Produccion	Show Column
Name	WebServer	Hide Column
aws:cloudformation:logical-id	WebServerInstance	Show Column
aws:cloudformation:stack-id	arn:aws:cloudformation:us-west-2:682086073548:stack/PruebaSSM/e2925c70-5578-11e9-9b29-066b98e74c72	Show Column
aws:cloudformation:stack-name	PruebaSSM	Show Column

<input type="checkbox"/>	Name	Ambiente	Instance ID	Instance Type
<input type="checkbox"/>	WebServer	Produccion	i-02e870bd8875a7dd6	t2.micro
<input checked="" type="checkbox"/>	WebTest	Pruebas	i-0e066f6bf4c715b9	t2.micro

Vamos a repetir este proceso en los siguientes recursos:

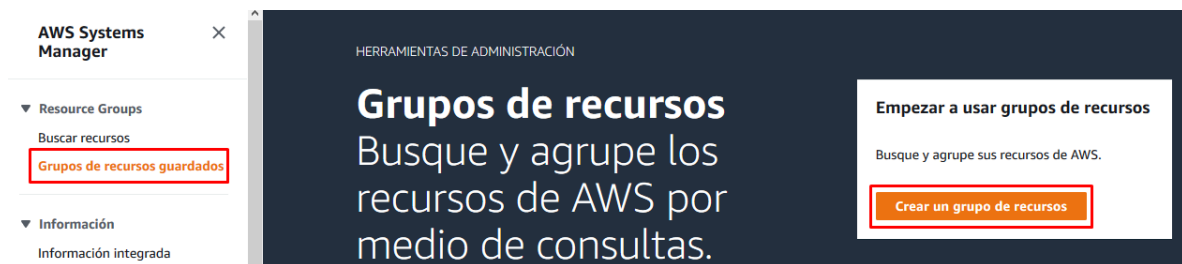
- Volúmenes (Separados producción y pruebas).

- Bucket S3 (solo producción).
- Security Group (solo producción).
- VPC (solo producción).
- Subnet (solo producción).
- Internet Gateway (solo producción).

Una vez configuradas todas las etiquetas, procedemos a crear los grupos de recursos.

Grupo de Recursos.

Nos dirigimos al servicio de AWS System Manager, buscándolo en el listado de servicios. Una vez abierta la consola de administración del System manager, vamos a la opción “grupo de recursos guardados”, y procedemos a crear un nuevo grupo de recursos:



Para crear el grupo de recursos, seleccionamos el grupo basado en etiquetas. Completamos las etiquetas para **Ambiente:Produccion** y damos clic en agregar. Esto se aprecia en la siguiente captura:

Luego, damos clic en “Ver recursos del grupo”:

Criterios de agrupación
Defina un grupo basado en los tipos de recursos y etiquetas.

Ver recursos del grupo

Resource types
Seleccionar tipos de recursos ▼

Todos los tipos de recursos admitidos

Tags

Tag key

Optional tag value

Add

Ambiente: Produccion X

Vemos entonces cómo AWS listará todos los recursos que tengan la etiqueta **Ambiente:Produccion**. Deben aparecer en total 7 recursos, como se aprecia a continuación:

Recursos del grupo (7)			
Q Buscar recursos			
Nombre	Servicio	Tipo	ID
S3 Bucket pruebassm-bucketpruebaicesissm-13fxquu8moccl 🔗	S3	Bucket	pruebassm-bucketpruebaicesissm-13fxquu8moccl
EC2 VPC vpc-09b2083e80b1f1423 🔗	EC2	VPC	vpc-09b2083e80b1f1423
EC2 SecurityGroup sg-0db358b00326fe76f 🔗	EC2	SecurityGroup	sg-0db358b00326fe76f
EC2 Volume vol-08051aec9d4655ef7 🔗	EC2	Volume	vol-08051aec9d4655ef7
EC2 Instance i-02e870bd8875a7dd6 🔗	EC2	Instance	i-02e870bd8875a7dd6
EC2 Subnet subnet-0b543eda48b2eddd0 🔗	EC2	Subnet	subnet-0b543eda48b2eddd0
EC2 InternetGateway igw-0d55a534f0271167f 🔗	EC2	InternetGateway	igw-0d55a534f0271167f

Para terminar, especificamos el nombre del grupo de recursos, su descripción y damos click en crear grupo.

Detalles del grupo

Nombre del grupo

128 caracteres como máximo. Debe empezar por una letra y solo puede contener letras, números y guiones.

Descripción del grupo - *opcional*

512 caracteres como máximo. Solo puede contener letras, números, guiones, caracteres de subrayado, puntos y espacios.

► **Etiquetas del grupo - *opcional***

Las etiquetas que especifique aquí no se aplicarán a los recursos del grupo, sino solo al propio grupo de recursos.

[Cancelar](#)
[Crear grupo](#)

Repetimos el proceso para la etiqueta **Ambiente:Pruebas**.

Al finalizar deberá tener creados los siguientes grupos de recursos:

Grupos de recursos

[Ver detalles](#)
[Crear grupo de recursos](#)

< 1 >
⚙️

	Nombre del grupo	Descripción
<input type="radio"/>	RecursosProduccion	Se agrupan los recursos de produccion del servidor Web.
<input type="radio"/>	RecursosPrueba	Grupo de recursos para prueba.

Información integrada.

El panel de información integrada, permite validar el estado de los recursos de AWS considerando los grupos de recursos creados previamente. En este menú, usted puede acceder a los reportes de los servicios: AWS Config, AWS CloudTrail, Personal Health Dashboard y Trusted Advisor. Para validar que los grupos de recursos funcionan correctamente, vaya a la pestaña del CloudTrail y valide que pueda ver los grupos de recursos creados:

- ▼ Insights
 - Built-In Insights
 - Dashboard by CloudWatch
 - Inventory
 - Compliance
- ▼ Actions
 - Automation
 - Run Command
 - Session Manager
 - Patch Manager

Config
CloudTrail
Personal Health Dashboard
Trusted Advisor

Resource Groups

Please select one resource group ▼

RecursosProduccion

Se agrupan los recursos de produccion del servidor Web.

RecursosPrueba

Grupo de recursos para prueba.

Resource ID	Resource type	Event time	Event name
No resource group is selected			

Seleccione alguno de los grupos de recursos y podrá visualizar los recursos del grupo y posteriormente los logs generados por el recurso seleccionado.

Resource Groups

RecursosProduccion ▼

< 1 >

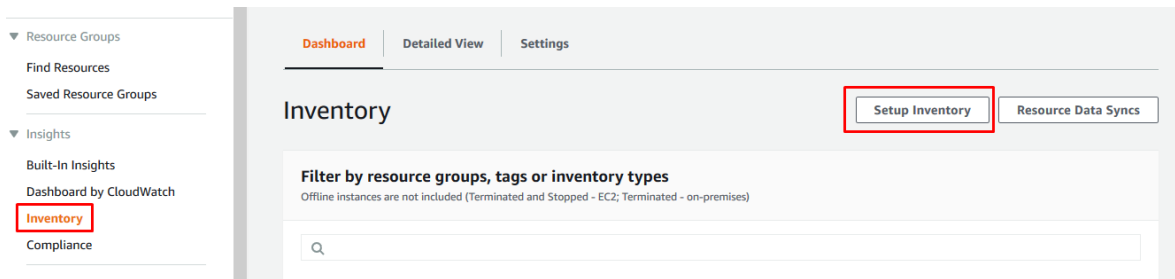
Resource ID	Resource type	Event time	Event name
pruebassm-bucketpruebaicesism-13fxquu8moccl	Bucket	2/4/2019 14:13:03	PutBucketTagging
vpc-09b2083e80b1f1423	VPC	2/4/2019 14:11:56	CreateTags
sg-0db358b00326fe76f	SecurityGroup	2/4/2019 14:09:23	CreateTags
vol-08051aec9d4655ef7	Volume	2/4/2019 14:07:25	CreateTags
i-02e870bd8875a7dd6	Instance	2/4/2019 14:03:44	CreateTags
subnet-0b543eda48b2eddd0	Subnet	€	€
igw-0d55a534f0271167f	InternetGateway	€	€

Filter:	Resource name ▼	i-02e870bd8875a7dd6	Time range:	Select time range	📅
	Event time	User name	Event name	Resource type	Resource name
▶	2019-04-02, 02:03:44 PM	JuanFelipeGomez	CreateTags		i-02e870bd8875a7dd6
▶	2019-04-02, 02:02:08 PM		AssumeRole	STS AssumedRole and 2 more	i-02e870bd8875a7dd6
▶	2019-04-02, 02:01:47 PM	JuanFelipeGomez	CreateTags		i-02e870bd8875a7dd6
▶	2019-04-02, 01:56:37 PM	JuanFelipeGomez	CreateTags		i-02e870bd8875a7dd6
▶	2019-04-02, 01:56:36 PM	JuanFelipeGomez	RunInstances	EC2 Instance and 6 more	i-02e870bd8875a7dd6
No more events					

Inventario.

Una de las mayores ventajas del System Manager es la posibilidad de tener un inventario de los recursos de AWS y los servicios internos, es decir, tener un inventario del software que tiene cada instancia. En nuestro laboratorio, no obtendremos mucha información, pero el procedimiento se aplica igual para cualquier infraestructura de AWS.

Vamos a la sección de Inventario, dentro de “in-sight”, y damos clic en la opción “*Setup Inventory*”.



Se nos abre una nueva ventana, donde cambiamos el nombre de la rutina de inventario y seleccionamos todos los recursos. Tengan en cuenta, que el inventario también lo puede realizar por recursos agrupados según una etiqueta, esto le permitirá obtener el inventario de recursos específicos y no de toda la infraestructura.

Setup Inventory

Create an inventory association to collect information about software and settings for a target set of managed instances.

Provide inventory details

Name - *Optional*

Provide a name for your Inventory.

Targets

Specify targets by

- ☒ Selecting all managed instances in this account
- ☐ Specifying a tag
- ☐ Manually selecting instances

Configuramos la programación en la que se va a actualizar el inventario de los recursos. Esto se puede hacer cada X días, horas o minutos. En nuestro laboratorio lo haremos cada 30 minutos. Adicionalmente, puede revisar que elementos desea que el agente de inventario obtenga de cada una de las instancias de AWS. En este caso, podemos dejar seleccionado todos los parámetros.

Schedule

(Requires SSMAgent version 2.0.790.0 and above)

Collect inventory data every

30



Minute(s) ▼

Parameters



Applications

(Optional) Collect data for installed applications.



AWS Components

(Optional) Collect data for AWS Components like amazon-ssm-agent.



Network Config

(Optional) Collect data for Network configurations.



Windows Updates

(Optional, Windows OS only) Collect data for all Windows Updates.



Instance Detailed Information

(Optional) Collect additional information about the instance, including the CPU model, speed, and the number of cores, to name a few.

Por último, podremos guardar los Logs generados en cada actualización de inventario en un bucket de S3. Especificamos el nombre del Bucket S3 y el prefijo de los Logs. En este caso utilizamos el buckets creado a través de nuestra plantilla y el prefijo puede ser cualquiera que ayude a identificar los logs.

Debe revisar bien el nombre de su bucket, ya que puede variar al que yo utilizo al momento de desarrollar esta guía. Finalmente, damos click en “Setup Inventory”

Advanced



Sync inventory execution logs to an S3 bucket

S3 bucket name

Type the name of a bucket in Amazon S3.

pruebassm-bucketpruebaicesism-13fxqu8moccl

S3 bucket prefix - optional

Type a prefix for the bucket above that receives the output; for example, mycommands/domainjoin.

Logs-Inventory-SSM

Cancel

Setup Inventory

Una vez todo correcto, seremos dirigidos a la ventana inicial, y damos clic en la pestaña de settings, para evaluar el estado del inventario. Al dar clic en el ID de la asociación, veremos el proceso de la rutina de actualización de inventario.

Dashboard	Detailed View	Settings
Inventory association		
<div> <input type="text"/> Setup new inventory Edit </div>		
<div> <input type="text"/> < 1 > </div>		
Association id	Association name	Last execution date
<input type="radio"/> 599e96de-b417-40ad-9a7a-c13df8d9623a	Inventory-WebLab	-
		Pending
		Resource status count
		Pending:2

Si todo funciona de manera correcta, deberá obtener el siguiente resultado:

Description	Resources	Parameters	Targets	Versions	Execution history
Document name	Association name				
AWS-GatherSoftwareInventory	Inventory-WebLab				
Document version	Association version				
\$DEFAULT	1				
Status	Association id				
Success	599e96de-b417-40ad-9a7a-c13df8d9623a				
Create date	Schedule expression				
Tue, 02 Apr 2019 20:08:13 GMT	rate(30 minutes)				
Last update association date	Last execution date				
Tue, 02 Apr 2019 20:08:13 GMT	Tue, 02 Apr 2019 20:08:16 GMT				
Output S3 bucket	Last successful execution date				
S3 Output	Tue, 02 Apr 2019 20:08:16 GMT				

Description	Resources	Parameters	Targets	Versions	Execution history
Resources					
< 1 >					
Resource id	Last applied on	Association status	Detailed status		
i-0e066f6bff4c715b9	Tue, 02 Apr 2019 20:08:16 GMT	✓ Success	View Output		
i-02e870bd8875a7dd6	Tue, 02 Apr 2019 20:08:16 GMT	✓ Success	View Output		

Al volver al menú del inventario, veremos que nos aparecen 2 servicios en estado “enable” y la información recopilada por el agente de inventario. Al final de la ventana, puede visualizar una tabla que resume la información más relevante de cada instancia. Esto se aprecia en las siguientes capturas:

Managed instances with inventory enabled

Includes instances in the current region and account.

Inventory coverage per type

Predefined Inventory Types only.

AWS:AWSComponent	100%
AWS:Application	100%
AWS:File	100%
nceDetailedInformation	100%
WS:InstanceInformation	100%
AWS:Network	100%
AWS:Service	100%
AWS:WindowsRegistry	100%
AWS:WindowsRole	100%
AWS:WindowsUpdate	100%

Top 10 Custom inventory types

Customer-defined inventory types for the inventory collection.

There is no data to display

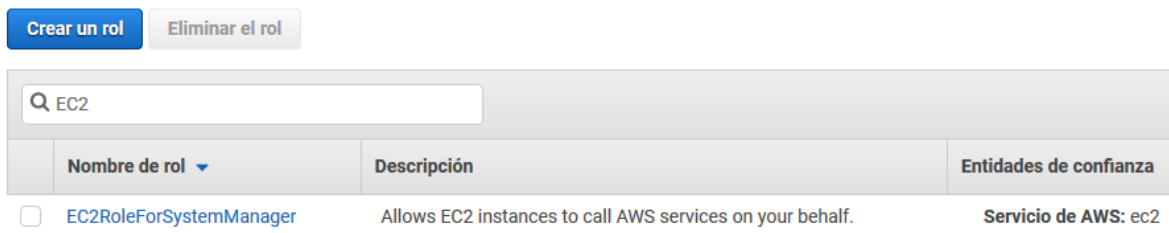
Corresponding managed instances

Instance ID	Name	Computer name	Platform type	Platform name
i-02e870bd8875a7dd6	WebServer	ip-10-0-0-122.us-west-2.compute.internal	Linux	Amazon Linux AMI
i-0e066f6bff4c715b9	WebTest	ip-10-0-0-141.us-west-2.compute.internal	Linux	Amazon Linux AMI

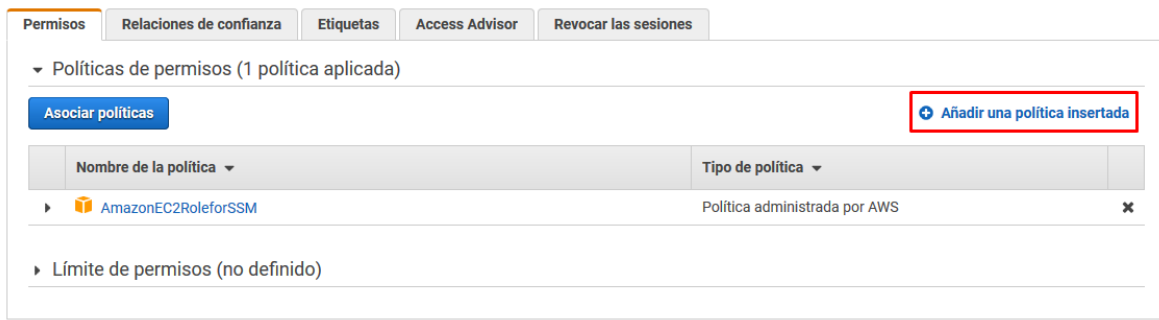
Session Manager.

Uno de los servicios que ayuda en la gestión de las instancias EC2 de AWS es el Session Manager. Session Manager es una funcionalidad de AWS Systems Manager completamente administrada que le permite gestionar instancias Amazon EC2 a través de un shell interactivo basado en navegador con un solo clic. Session Manager proporciona una administración de instancias segura y auditable sin la necesidad de abrir los puertos de entrada o administrar claves SSH.

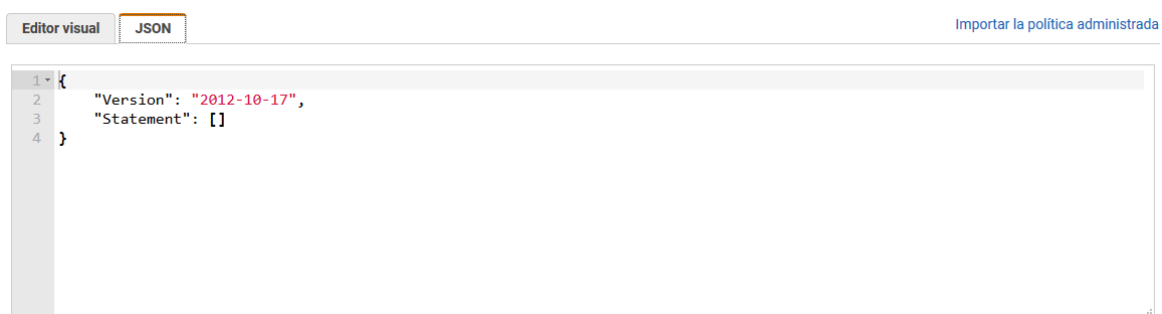
Para garantizar que el servicio de System Manager pueda iniciar una consola de administración a través de un Shell, debemos agregar permisos adicionales al rol de EC2 creado al inicio de esta guía.



Una vez dentro del rol, damos clic en la opción “Añadir una política insertada”.



Cambie a la pestaña de JSON.



Copie y pegue el siguiente fragmento de código:

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

```

"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "ssm:UpdateInstanceInformation",
      "ssmmessages:CreateControlChannel",
      "ssmmessages:CreateDataChannel",
      "ssmmessages:OpenControlChannel",
      "ssmmessages:OpenDataChannel"
    ],
    "Resource": "*"
  },
  {
    "Effect": "Allow",
    "Action": [
      "s3:GetEncryptionConfiguration"
    ],
    "Resource": "*"
  }
]
}

```

Luego de clic en “*revisar política*” y asignamos un nuevo nombre a esta política: EC2SessionManager.
Damos clic en crear política.

Nombre*

128 caracteres como máximo. Utilice caracteres alfanuméricos y "+,=, @, _".

Resumen

Filtro:			
Servicio	Nivel de acceso	Recurso	Condición de solicitud
Permitir (2 de 174 servicios) Mostrar 172 restantes			
S3	Limitado: Lectura	Todos los recursos	Ninguna
SSM Messages	Acceso completo	Todos los recursos	Ninguna

[Cancelar](#)

[Anterior](#)




[Crear una política](#)

Ahora nuestro rol debe tener las siguientes políticas asociadas:


Permisos Relaciones de confianza Etiquetas Access Advisor Revocar las sesiones



Políticas de permisos (2 políticas aplicadas)



[Asociar políticas](#) [+ Añadir una política insertada](#)


Nombre de la política	Tipo de política	
 AmazonEC2RoleforSSM	Política administrada por AWS	
EC2SessionManager	Política insertada	

Con esta configuración ya podremos ir a la ventana de sesión manager y dar click en “Start Session”. Luego, nos aparecerán las instancias asociadas al SystemManager, seleccionamos la instancia de pruebas y nos deberá aparecer el siguiente mensaje:

Target instances 

 1 

	Instance name	Instance ID	Agent version	Instance state	Availability zone	Platform
<input checked="" type="radio"/>	WebServer	i-02e870bd8875a7dd6	2.2.916.0	 running	us-west-2b	Amazon Linux AMI
<input type="radio"/>	WebTest	i-0e066f6bffc715b9	2.2.916.0	 running	us-west-2b	Amazon Linux AMI

 The version of SSM Agent on the instance does not support Session Manager. Update the agent to the latest version. [Learn more](#)

[Update SSM Agent](#)


[Cancel](#) [Start session](#)

Esto se debe a que durante la creación de las instancias utilizamos un AMI con la versión del agente del System Manager desactualizada. Para solucionar estos problemas damos clic en la opción de actualizar el SSM Agent a la versión más reciente.

Aquí AWS, utilizará otra de las herramientas del System Manager, llamada Run Command. Run Command nos permite ejecutar una tarea simple que se ejecuta una única vez. En este caso, se ejecutará la tarea de actualización del agente SSM sobre la instancia.



Command ID: 3c69c98e-da19-494c-ad11-01bb74d89958  [Cancel command](#)

Command status

Overall status	# targets	# completed	# error	# delivery timed out
 Success	1	1	0	0

Targets and outputs [View output](#)

< 1 >

Instance ID	Instance name	Status	Start time	Finish time
 i-02e870bd8875a7dd6		 Success	Tue, 02 Apr 2019 21:05:50 GMT	Tue, 02 Apr 2019 21:06:03 GMT

Ahora volvemos al Session Manager y volveremos a intentar iniciar una sesión en la instancia de pruebas. Veremos entonces que se nos abre una nueva pestaña en el navegador con la consola del servidor. Aquí ya podremos ejecutar los comandos y tareas que necesites para configurar y dar mantenimiento al servidor.

Session ID: JuanFelipeGomez-0b4d87ad9c1dfc8e4 Instance ID: i-02e870bd8875a7dd6 [Terminate](#)

```
sh-4.2$ ls
bin  boot  cgroup  dev  etc  home  lib  lib64  local  lost+found  media  mnt  opt  proc  root  run  sbin  selinux  srv  sys  tmp  usr  var
sh-4.2$ ifconfig
eth0      Link encap:Ethernet  HWaddr 02:AA:41:3C:2D:B2
          inet addr:10.0.0.122  Bcast:10.0.0.255  Mask:255.255.255.0
          inet6 addr: fe80::aa:41ff:fe3c:2db2/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:9001  Metric:1
          RX packets:88900 errors:0 dropped:0 overruns:0 frame:0
          TX packets:29964 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:115790411 (110.4 MiB)  TX bytes:3730693 (3.5 MiB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:2 errors:0 dropped:0 overruns:0 frame:0
          TX packets:2 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:140 (140.0 b)  TX bytes:140 (140.0 b)

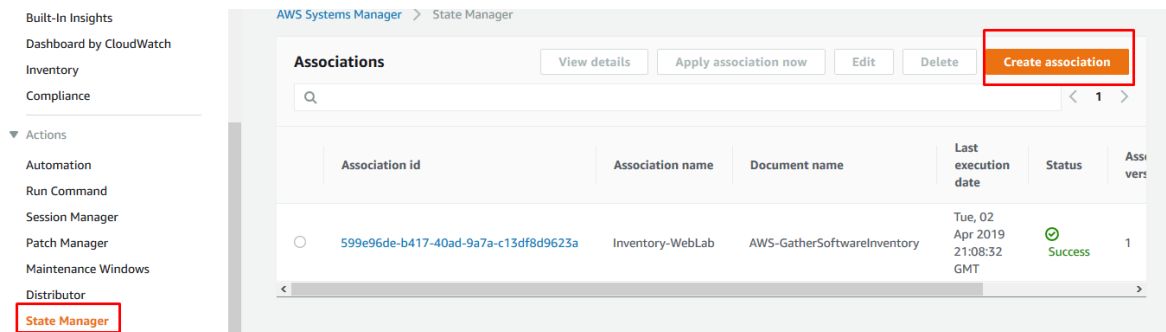
sh-4.2$
```

Para terminar la sesión damos clic en Terminate.

State Manager.

Cómo puede darse cuenta, actualizar una a una cada instancia que tenga el agente del System Manager desactualizado es una tarea engorrosa que puede ser automatizada a través del administrador de estado de AWS (State Manager). Accedemos al state manager y vemos que tenemos ya una asociación creada. Dicha asociación hace referencia a la rutina que se encarga de actualizar el inventario.

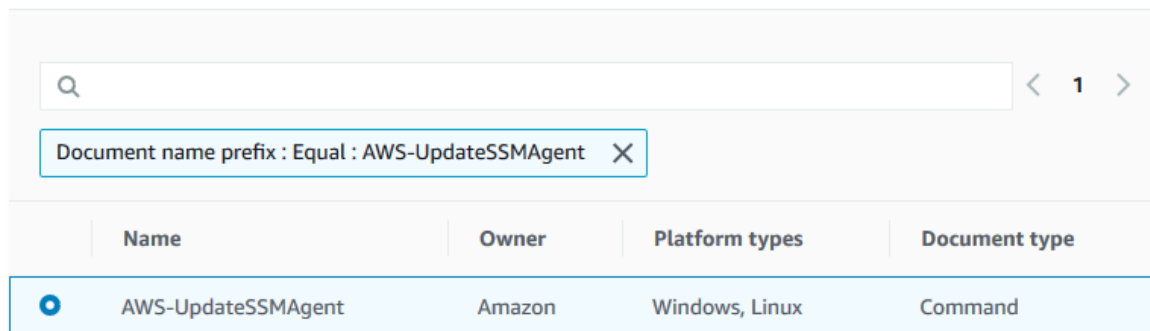
Procedemos entonces a crear una nueva asociación de estado, dando clic en la opción “*Create association*”



Damos el nombre que deseemos a la asociación y luego veremos una lista de funciones que podemos utilizar. Estas funciones son realizadas por el equipo de AWS y definen el conjunto de actividades que el system manager debe desarrollar para completar una tarea automatizada. Encontramos tareas para gestión de volúmenes, instancias, buckets, bases de datos, snapshot. Básicamente, todo lo que pueda ser automatizado por AWS que sea una tarea rutinaria se encuentra en el listado de funciones que nos presentan en el state manager.

Estas funciones posibilitan que la gestión centralizada y automatizada de infraestructura de AWS sea realmente muy sencilla de realizar. Es importante aclarar que las funciones se encuentran por orden alfabético.

Buscamos entonces la función: **AWS-UpdateSSMAgent**. Como se aprecia en la siguiente captura:



En los parámetros, dejamos la versión sin datos y no permitimos que sea posible hacer downgrade a la versión del agente del System Manager.

Parameters

Version

(Optional) A specific version of the Amazon SSM Agent to install. If not specified, the agent will be updated to the latest version.

Allow Downgrade

(Optional) Allow the Amazon SSM Agent service to be downgraded to an earlier version. If set to false, the service can be upgraded to newer versions only (default). If set to true, specify the earlier version.

Es importante seleccionar que se actualicen todas las instancias.

Targets

Targets are the instances you would like to associate with this document. You can choose to target by both managed instance and tag.

Select targets by

- ☒ Selecting all managed instances in this region under this account
- ☐ Specifying tags
- ☐ Manually Selecting Instance

Ahora bien, programe la función para que se ejecute todos los días a las 00:00 UTC. Puede elegir también que se ejecute cada domingo, para evitar afectaciones en el

Specify schedule

On Schedule ☒

Run association at cron/rate intervals.

No schedule ☐

Run association once.

Specify with

- ☒ CRON schedule builder
- ☐ Rate schedule builder
- ☐ CRON/Rate expression

Association runs

☐ Every 30 minutes

☐ Every Hour

☒ Every at :

Por último, configure los parámetros del bucket S3 para guardar los Logs y damos clic en crear asociación. Esto creara nuestra rutina automatizada.

Output options

Write to S3
Write all command output to an Amazon S3 bucket. Command output in the console is truncated after 2500 characters.

☒ **Enable writing output to S3**

S3 bucket name
Specify the name of your bucket.


S3 key prefix - optional
Type a prefix for the bucket that receives the output; for example, mycommands/domainjoin.

[Cancel](#) [Create Association](#)

Al finalizar si todo sale bien deberá obtener el siguiente resultado.

Association ID: bce14ef6-da5c-49d3-83c9-2e21b4c6883e

[Apply association now](#) [Edit](#) [Delete](#)

Description	Resources	Parameters	Targets	Versions	Execution history
Document name AWS-UpdateSSMAgent				Association name ActualizarSSMAgent	
Document version \$DEFAULT				Association version 1	
Status  Success				Association id bce14ef6-da5c-49d3-83c9-2e21b4c6883e	
Create date Tue, 02 Apr 2019 21:44:26 GMT				Schedule expression cron(0 00 00 ? * *)	

AWS Systems Manager > State Manager > Association ID : bce14ef6-da5c-49d3-83c9-2e21b4c6883e > Resources

Association ID: bce14ef6-da5c-49d3-83c9-2e21b4c6883e

[Apply association now](#)
[Edit](#)
[Delete](#)

[Description](#)
[Resources](#)
[Parameters](#)
[Targets](#)
[Versions](#)
[Execution history](#)

Resources

< 1 >

Resource id	Last applied on	Association status	Detailed status
i-0e066f6bff4c715b9	Tue, 02 Apr 2019 21:44:40 GMT	✓ Success	View Output
i-02e870bd8875a7dd6	Tue, 02 Apr 2019 21:44:28 GMT	✓ Success	View Output

Para configurar cualquier función del administrador de estado del System manager puede seguir los mismos procedimientos de esta guía. Lo único que cambia es el campo de **Parametros**.

Start/Stop Instance.

Una manera en la que puede aprovecharse el System Manager es la programación del apagado e inicio de las instancias EC2. Para realizar esta tarea es importante tener en cuenta que, al momento de lanzar un stop en las instancias, se libera la dirección IP Pública. Para evitar esto, y dejar la misma dirección IP pública en todo momento, se debe configurar el servicio Elastic IP.

Para configurar la IP elástica vamos a los servicios de VPC.

VPC Dashboard

Filter by VPC:

Virtual Private Cloud

Your VPCs

Subnets

Route Tables

Internet Gateways

Egress Only Internet Gateways

DHCP Options Sets

Elastic IPs

Endpoints

Endpoint Services

Launch VPC Wizard **Launch EC2 I**

Note: Your Instances will launch in the US East (N. Virgir

Resources by Region [Refre](#)

You are using the following Amazon VPC resources

VPCs N. Virginia **7**
[See all regions](#)

Subnets N. Virginia **23**
[See all regions](#)

Route Tables N. Virginia **17**
[See all regions](#)

Internet Gateways N. Virginia **7**
[See all regions](#)

Buscamos una nueva dirección IP pública.

Allocate new address **Actions** ▾

<input type="checkbox"/>	Name	Elastic IP	Allocation ID	In
<input type="checkbox"/>		52.21.205.217	eipalloc-0f977649...	-

Seleccionamos las IPs del Pool de Amazon y damos click en encontrar nueva dirección IP.

Allocate new address

Allocate a new Elastic IP address by selecting the scope in which it will be used

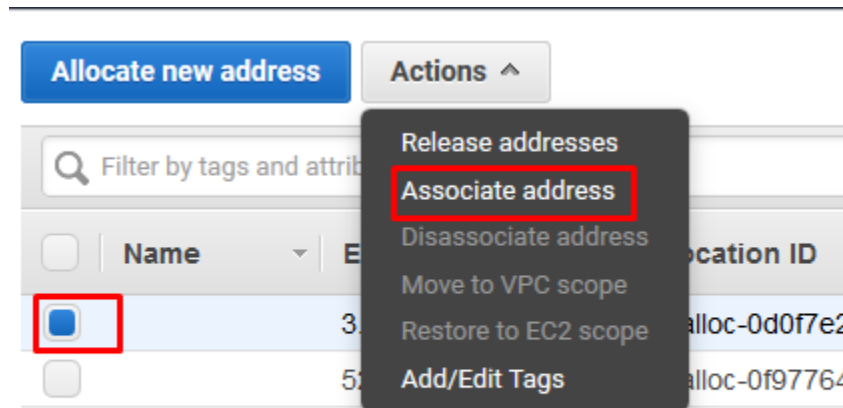
Scope VPC

IPv4 address pool ☒ Amazon pool
☐ Owned by me

* Required

[Cancel](#) **Allocate**

Una vez tengamos una nueva dirección IP, procedemos a amarrarla a la instancia que hemos denominado cómo la de producción.



Associate address

Select the instance OR network interface to which you want to associate this Elastic IP address (3.88.98.223)

Resource type ☒ Instance **i** ☐ Network interface

Instance **↻**

Private IP **i**

Reassociation

Instance ID	Name	State
i-04754d7c6dbdae9a3	Web_Produccion	running
i-0d857058ffa8e4150	Web_Test	running

La dirección IP privada no presenta cambios y se usará la misma. Por último validamos que no se permita re asociar la IP elástica mientras se encuentre asociada a una instancia EC2.

Reassociation ☐ Allow Elastic IP to be reassociated if already attached **i**

Una vez asociada la IP tendremos el siguiente resultado:

Filter by tags and attributes or search by keyword						
<input type="checkbox"/>	Name	Elastic IP	Allocation ID	Instance	Private IP address	Scope
<input checked="" type="checkbox"/>		3.88.98.223	eipalloc-0d0f7e29...	i-04754d7c6dbda...	10.0.0.238	vpc

Name	Instance ID	Instance Type	Availability Zone	Instance State	Status Checks	Alarm Status	Public DNS (IPv4)	IPv6
Web_Produc...	i-04754d7c6dbdae9a3	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-3-88-98-223.compu...	3.8
Web_Test	i-0d857058ffa8e4150	t2.micro	us-east-1f	running	2/2 checks ...	None	ec2-18-207-238-241.co...	18.1

Instance: i-04754d7c6dbdae9a3 (Web_Produccion) Elastic IP: 3.88.98.223

Description

Status Checks

Monitoring

Tags

Instance ID

i-04754d7c6dbdae9a3

Instance state

running

Instance type

t2.micro

Elastic IPs

3.88.98.223*

Availability zone

us-east-1f

Security groups

[PruebaRedHat-WebServerSecurityGroup-1HZCFJMXM6OM6](#), [view inbound rules](#), [view outbound rules](#)

Scheduled events

No scheduled events

AMI ID

amzn-ami-hvm-2018.03.0.20180811-x86_64-

Public DNS (IPv4)

ec2-3-88-98-223.compute-1.amazonaws.com

IPv4 Public IP

3.88.98.223

IPv6 IPs

-

Private DNS

ip-10-0-0-238.ec2.internal

Private IPs

10.0.0.238

Secondary private IPs

VPC ID

vpc-0a258f09ac433a95e

Subnet ID

subnet-0dc28df8b6054b5c9

Ahora bien, vamos a la opción de State Manager dentro del System Manager y seleccionamos crear nueva asociación.

Management tools

AWS Systems Manager

State Manager

Cross-platform, fleet-wide configuration management solution

Centrally manage the configurations of your Amazon EC2 Windows and Linux instances and your on-premises servers or virtual machines (VMs).

Create a configuration

Create an Association

Use Cases and Blogposts

Nombramos la asociación y buscamos el documento AWS-StopEC2Instance

	Name	Owner	Platform types	Document type
<input type="radio"/>	AWS-RunShellScript	Amazon	Linux	Command
<input type="radio"/>	AWS-SetupInventory	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-SetupManagedInstance	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-SetupManagedRoleOnEc2Instance	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-StartEC2Instance	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-StartEC2InstanceWithApproval	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-StartRdsInstance	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-StopEC2Instance	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-StopEC2InstanceWithApproval	Amazon	Windows, Linux	Automation
<input type="radio"/>	AWS-StopRdsInstance	Amazon	Windows, Linux	Automation

Ahora bien, seleccionamos las instancias objetivo. En este caso serán las 2 instancias que estamos utilizando en el laboratorio. En el Rol de automatización usamos el Rol creado durante este laboratorio, AutomationServiceRole

Show managed instances only ▼

< 1 >

<input checked="" type="checkbox"/>	Name	Instance ID	State	Availability zone
<input checked="" type="checkbox"/>	Web_Produccion	i-04754d7c6dbdae9a3	running	us-east-1f
<input checked="" type="checkbox"/>	Web_Test	i-0d857058ffa8e4150	running	us-east-1f

AutomationAssumeRole

(Required) The ARN of the role that allows Automation to perform the actions on your behalf.

AutomationServiceRole ▼

Por último, programamos la ejecución de la tarea con el constructor de CRON. La idea es que las máquinas se detengan en 5 minutos. Es importante tener en cuenta el CRON maneja horario UTC, porque que tendrá que ajustar el horario a la zona de Colombia (UTC-5)

En la siguiente captura de pantalla vemos cómo se programa el apagado de las instancias cada día a las 02:30 AM (UTC), es decir, 09:30 PM hora colombia

Specify schedule

On Schedule ☒

Run association at cron/rate intervals.

No schedule ☐

Run association once.

Specify with

☒ CRON schedule builder

☐ Rate schedule builder

☐ CRON/Rate expression

Association runs

☐ Every 30 minutes

☐ Every Hour

☒ Every at :

Cancel

Create Association

Puede realizar exactamente el mismo procedimiento anterior con la función AWS-StartEC2Instance.

Bibliografía

Amazon. (s.f.). *¿Qué es AWS Systems Manager?* Obtenido de https://docs.aws.amazon.com/es_es/systems-manager/latest/userguide/what-is-systems-manager.html