云享家 | 基于SSM+Chef InSpec的操作系统合规审计

BespinGlobal BespinGlobal 9月29日

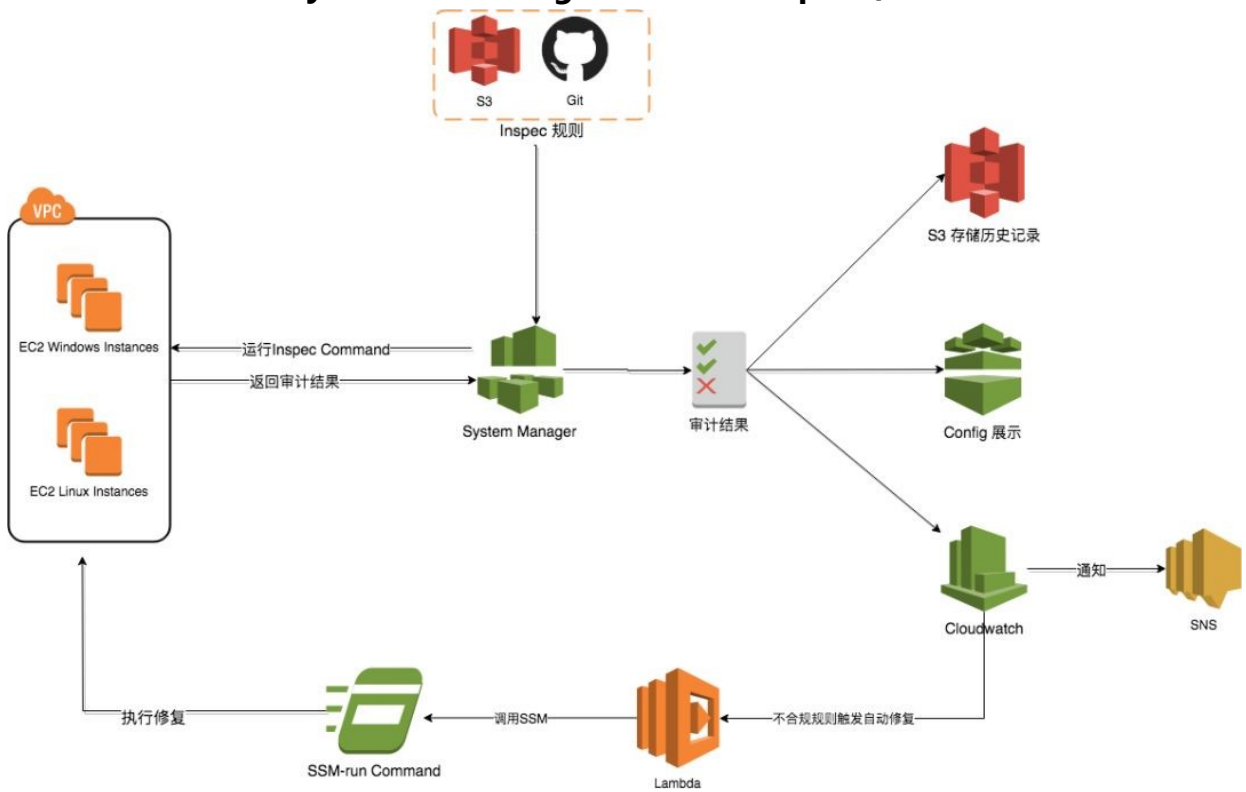
基于SSM+Chef InSpec的操作系统合规审计

1 为什么要做系统合规审计?

是否文件的权限设置过大？是否数据库的端口监听在0.0.0.0？是否Swap没有开启？是否通过手动的方式检测系统确保系统的种种配置符合内部策略的定义？此方案使用Chef InSpec将您的合规性表示成代码，通过AWS Systems Manager执行自动化和批量的操作，对操作系统进行全面扫描，确保操作系统的合规性。

2 操作系统合规架构

2.1. 基于AWS Systems Manager+Chef Inspec架构



2.2. 架构说明

AWS Systems Manager服务为核心，运行Chef InSpec扫描，Systems Manager接收扫描结果，通过CloudWatch对结果触发SNS通知，也可以将事件发送至Lambda，Lambda返回调用Systems Manager RunCommand执行自动修。

2.3. 使用的服务

- Systems Manager: 大规模的配置和管理EC2实例，也支持管理非EC2的虚拟机，通过在虚拟机安装代理，执行自动化操作。
- S3: 存储Systems Manager运行的结果。

- CloudWatch: 检测Systems Manager的状态，并触发之后的操作，包括触发SNS通知，触发Lambda，Lambda调用Systems Manager Run Command模块执行操作系统中的命令进行不合规的修复。
- SNS: 评估结果的通知，可以支持邮件及HTTP终端节点。
- Chef InSpec: 是一个开源框架，能够进行操作系统和应用程序的审核

2.4. 实现功能

- 自动发现: AWS Systems Manager可以通过指定Tags的方式对指定的EC2资源进行操作，如果新创建虚拟机，则只需要配置指定的标签，则Systems Manager会将其加入扫描列表里；
- 持续审核与合规: 通过Chef InSpec规则，评估目前的所有资源及之后资源的更改是否符合内部策略要求和监管标准；
- 合规性即代码: 通过Chef InSpec和Systems Manager结合，自定义规则，将合规性按要求编制成代码，从而将内部策略要求以自动的方式确保AWS的基础设施的合规性和自动修复；
- 故障排除: 查看Systems Manager获得的扫描结果，查看资源最近配置更改，实现快速故障排除操作问题；
- 安全性分析: 通过Chef InSpec规则，审计配置的安全弱点
- 自动化: 通过Systems Manager, 可以配置批量部署，定时执行，批量扫描，并通过预先写好的脚本，批量自动修复，全流程实现自动化操作。
- 自动修复: 通过CloudWatch事件，检测Systems Manager的合规状态，CloudWatch将事件发送至Lambda,Lambda调用Systems Manager 对不合规的规则执行自动修复的命令。

3 中国区使用注意事项

- 在Global, CloudWatch事件能够触发SSM RunCommand, 中国区目前不知此服务，但中国区CloudWatch能够触发Lambda，因此通过Lambda调用SSM 执行修复操作
- Inspec规则可以存储在Github上，SSM Document可以指定Checkout Git上代码，但是默认的Inspec文档要求存储Git的token在KMS下面，但是中国区目前没有KMS服务，因此，我们将规则存储在S3上进行调用

以下介绍方案中的一部分，如何通过SSM运行Inspec(中国区)

4 SSM运行Inspec创建过程

4.1 创建Role

注意: 如果您的EC2实例已经附加了Role,则无需创建Role及附加Role,只需要为该Role增加" AmazonEC2RoleforSSM"策略即可。

1）进入到AWS IAM控制台，进入到"角色"，点击"创建角色"

2）在策略下面选择策略"AmazonEC2RoleforSSM":

3）填写Role的名称。

4）点击"创建角色",等待创建完成。

4.2 附加Role

1）进入到EC2规则控制台页面下，选定您的EC2实例，

- 选择"操作"à"实例设置"à"附加/更换IAM角色"

2）选择之前所创建的Role，点击"应用"。

4.3 安装SSM Agent

注意: 默认使用Amazon Linux镜像，镜像已经包含安装了的SSM Agent，则可用跳过该步骤，如果使用非Amazon Linux镜像，以下为Centos7的安装过程。其它系统的SSM Agent的安装过程，请参考AWS官方文档(https://docs.aws.amazon.com/zh_cn/systems-manager/latest/userguide/ssm-agent.html)

1) 登录EC2实例

2) 创建临时目录

```
mkdir /tmp/ssm
cd /tmp/ssm
```

3）运行安装

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

4）启动服务

```
sudo systemctl enable amazon-ssm-agent
sudo systemctl start amazon-ssm-agent
```

5）检查状态

```
sudo systemctl status amazon-ssm-agent
```

6）完成之后，等待一会儿，在SSM控制台"托管实例"下面即可看到该实例，表明该实例已在SSM管理之下，如下图所示 (托管实例下面无需做任何配置，一旦您的EC2实例绑定好具有SSM权限的Role,并启动了SSM Agent，则会自动显示在托管实例下面)

4.4 安装Chef Development Kit

注意: SSM RunInspec文档默认检测系统未安装Chef DK,会自动安装，但是其文档安装是使用Chef官方的源，并且由于Chef DK包较大，受网络影响，安装过程极不稳定，同时，由SSM安装的包，在其运行完之后会卸载其安装的Chef DK，为了不让其卸载，现提前手动安装好Chef DK

- 从Chef官网下载Chef DK的包，https://downloads.chef.io/chefdk，选择对应的操作系统的版本下载安装
- 安装完成之后，进行检查

```
# inspec –version
```

4.5 创建S3 Bucket并上传Inspec代码

1）在AWS S3控制台创建S3 Bucket

注意: SSM只能从同一Region的S3 Bucket获取资源，因此Bucket需要在您EC2运行所在的区域创建。

2）从Gitlab上下载Inspec规则，并上传至S3 上，我们以检测Linux合规为例，以下为linux-baseline的inspec代码

https://github.com/dev-sec/linux-baseline

4.6 创建SSM关联

1）进入EC2 控制台，点击"状态管理器"，点击"创建关联"

2）输入关联名称，选择”AWS-RunInspecChecks”文档。

3）选择实例，及计划时间，并在参数部分，填写之前上传的Inspec规则的S3地址

- 您也可以选择通过标签的方式选择实例
- 在参数部分选项S3,填写S3地址，S3地址格式如下:

举例，我们之前将linux-baseline上传到bgc-inspec-check-bj存储桶下，则填写内容截图如下所示:

4）开启日志写入到S3， 输入存储桶区域，名称及前缀，点击“创建关联”，等待创建完成(可选)。

5）等待创建完成之后，在“状态管理器”界面下，可以看到刚刚所创建的关联，选中该关联，点击“立即应用关联”，则SSM开始运行。

6）等待运行结束，则可看到合规结果,在“托管实例”下“配置合规性”即可看到合规结果。

7）接下来，SSM会按照之前所设定计划时间定时执行。

参考:

- AWS Global 博客

  https://aws.amazon.com/cn/blogs/mt/using-aws-systems-manager-to-run-compliance-scans-using-inspec-by-chef/

- Dev Sec参考规则

  https://github.com/dev-sec/

- AWS SSM 合规官方文档

  https://docs.aws.amazon.com/zh_cn/systems-manager/latest/userguide/systems-manager-compliance.html

阅读原文