

1. Las estrategias de Chaos Monkey son:
  - i. **Shutdown instance (Simius Mortus):** Apaga la instancia utilizando el API de EC2
  - ii. **Block all network traffic (Simius Quies):** Remueve todos los grupos de seguridad de la instancia y la mueve a un grupo alterno que no permite accesos, bloqueando toda conexión por medio de la red a pesar de seguir en funcionamiento
  - iii. **Detach all EBS volumes (Simius Amputa):** Desata todos los volúmenes de EBS de la instancia, simulando una falla de EBS.
  - iv. **SSH Monkeys:** Todos los monkeys restantes trabajan corriendo scripts sobre la instancia. Para configurarse es necesario definir la llave SSH de logeo a la instancia
  - v. **Burn-CPU (Simius Cogitarius):** Corre procesos intensivos de CPU, simulando un vecino ruidoso o fallos de CPU
  - vi. **Burn-IO (Simius Occupatus):** Corre procesos intensivos de disco, simulando un vecino ruidoso o fallos de disco
  - vii. **Fill Disk (Simius Plenus):** Escribe un archivo inmenso sobre el dispositivo raíz, llenando el (relativamente pequeño) disco raíz de EC2
  - viii. **Kill Processes (Simius Delirius):** Mata cada programa de Java o Python que encuentra cada segundo, simulando una aplicación con fallos, instalación corrupta o instancia fallida
  - ix. **Null-Route (Simius Desertus):** Anula la ruta de red 10.0.0.0/8 usada por la red interna de EC2, de manera que todo el tráfico EC2 <-> EC2 fallará
  - x. **Fail DNS (Simius Nonomenius):** Utiliza las tablas IP para bloquear el puerto 53 para TCP y UDP, los puertos para tráfico de DNS, simulando fallas del servidor DNS
  - xi. **Fail EC2 API (Simius Noneccius):** coloca entradas host dummy dentro de /etc/hosts, de manera que el API de comunicación EC2 falle simulando un problema de plano de control en EC2 (En caso de que el servicio no sea utilizado por la instancia, no tiene efecto alguno)
  - xii. **Fail S3 API (Simius Amnesius):** coloca entradas host dummy dentro de /etc/hosts, de manera que el API de comunicación S3 falle simulando un problema de plano de control en S3 (En caso de que el servicio no sea utilizado por la instancia, no tiene efecto alguno)
  - xiii. **Fail DynamoDB API (Simius Nodynamus):** coloca entradas host dummy dentro de /etc/hosts, de manera que el API de comunicación DynamoDB falle simulando un problema de plano de control en DynamoDB (En caso de que el servicio no sea utilizado por la instancia, no tiene efecto alguno)
  - xiv. **Network Corruption (Simius Poiliticus):** Utiliza el API de modelado de tráfico para corromper una gran fracción de paquetes de red, simulando una degradación en la red EC2
  - xv. **Network Latency (Simius Tardus):** Utiliza el API de modelado de tráfico para introducir latencia (1 segundo +-50%) a todos los paquetes de red, simulando una degradación en la red EC2
  - xvi. **Network Loss (Simius Perditus):** Utiliza el API de modelado de tráfico para soltar una fracción de los paquetes de red, simulando una degradación en la red EC2

2. El Janitor Monkey se encarga de limpiar todos los recursos que se cree no están siendo usados mediante un conjunto de reglas. Las reglas determinan el tipo de archivos o recursos que deberían ser limpiados, de manera que una vez el Monkey pasa por ahí los marca y asigna una fecha de borrado.

Como es posible que algunos casos excepcionales no hayan sido contemplados dentro de las reglas el Monkey notifica cuando un recurso es marcado, de manera que es posible re-marcarlo como una excepción y en este caso sería ignorado al momento de la eliminación.

3. El Conformity Mokey se encarga de marcar y notificar en caso de encontrar instancias que no cumplen con un conjunto de reglas definidas para mejores prácticas. Sin embargo, es posible configurarlo para que ignore determinadas instancias o grupos en caso de no ser necesaria una notificación por parte de este.