Escenario para Pentesting (I). Introducción y objetivos (WIP)

2020-09-15

Se va a diseñar un escenario virtual que emule una infraestructura de una empresa de cierto tamaño, aunque para reducir el uso de recursos necesarios para el despliegue del escenario, se van a juntar los servicios que tienen coherencia en la misma máquina, haciendo por ejemplo que distintas bases de datos estén en el mismo host.

Los principales objetivos para el desarrollo de este escenario son los siguientes:

- Estudio en profundidad de los principales servicios usados en la actualidad, ya que va a ser necesario instalarlos y configurarlos adecuadamente.
- Creación de un escenario (basic) con las distintas configuraciones de los servicios por defecto a excepción de los logs, que serán enviados a un servidor centralizado de logs, lo que nos permitirá posicionarnos desde dos puntos de vista diferentes:
  - Red Team: que tiene como objetivo realizar un estudio de estas configuraciones buscando fallos de seguridad en estas para posteriormente realizar ataques aprovechándose de esas configuraciones por defecto.
  - Blue Team: Al mismo tiempo, desde este punto de vista realizaremos un análisis de los logs para ver que rastros se dejan durante el ataque.
- Creación de un escenario (advanced) al que se le van a aplicar configuraciones adecuadas según las necesidades de cada servicio desplegado. En esta etapa también se probará a realizar distintos tipos de ataque y analizar los logs para ver los registros generados.
- Creación de un escenario (hardening) en el que todas las maquinas y servicios van a tener aplicadas configuraciones más robustas siguiendo estándares de guías como el CIS o el NIST.

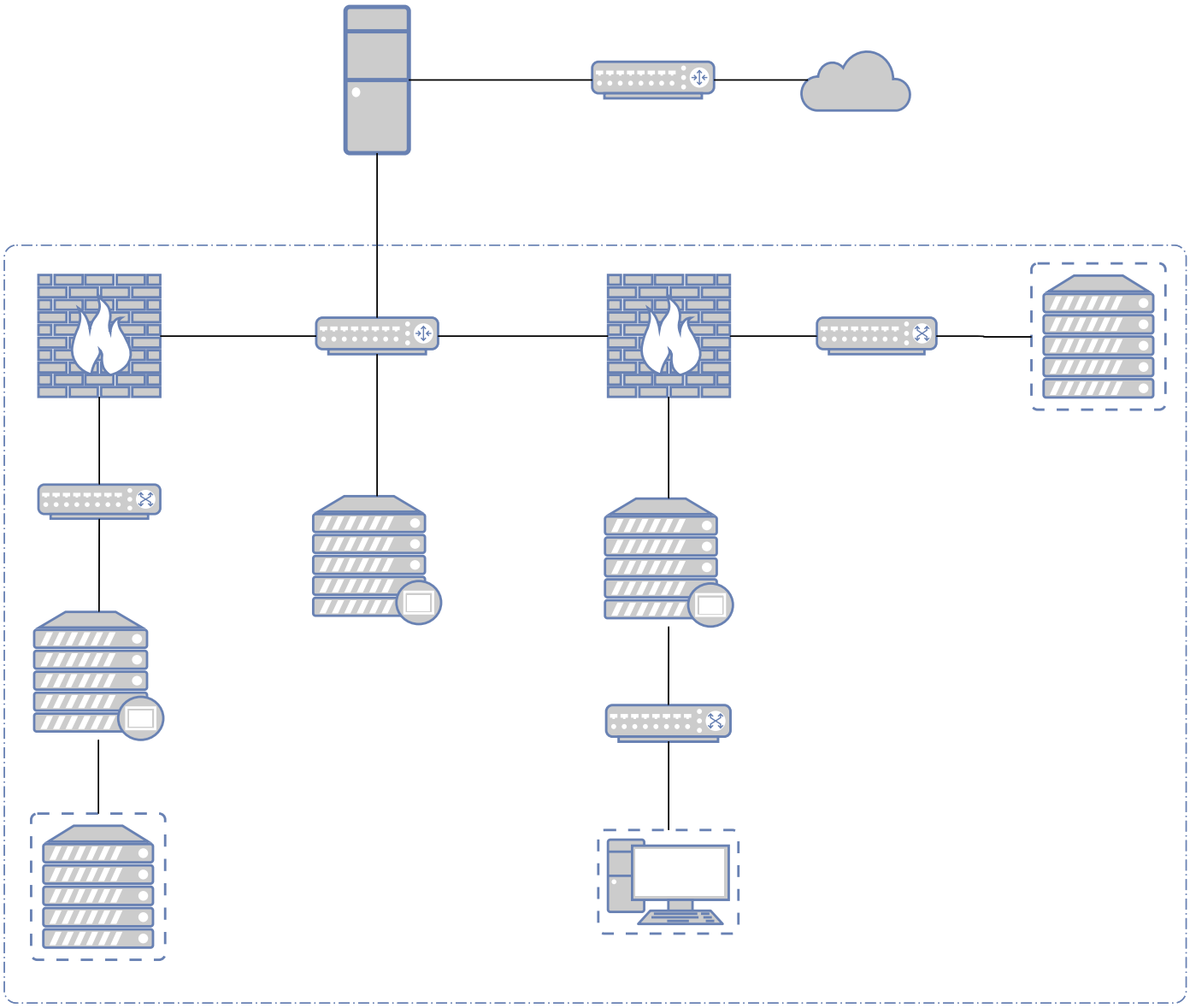
Las principales herramientas que se van a utilizar para el despliegue del escenario son las siguientes:

- **Proxmox**: Proxmox Virtual Environment, o Proxmox VE, es un entorno de virtualización de servidores de código abierto. Está en distribuciones GNU/Linux basadas en Debian con una versión modificada del Kernel RHEL y permite el despliegue y la gestión de máquinas virtuales y contenedores. Proxmox VE incluye una consola Web y herramientas de línea de comandos, y proporciona una API REST para herramientas de terceros. Dos tipos de virtualización son compatibles: los contenedores basados con LXC y la virtualización con KVM. Viene con un instalador e incluye un sitio Web basado en la interfaz de administración.
- **Terraform**: Terraform es un software de infraestructura como código (IaaC) desarrollado por HashiCorp. Permite a los usuarios definir y configurar la infraestructura de un centro de datos en un lenguaje de alto nivel, generando un plan de ejecución para desplegar la infraestructura en Proxmox, OpenStack, AWS, Azure, etc.
- **Bash**: Para la automatización de configuración y despliegue de Proxmox, hosts y los distintos contenedores se van a utilizar scripts hechos en bash que realicen una configuración inicial.
- **Ansible**: Para la instalación y configuración de los servicios en todos los hosts. El objetivo es crear distintos entornos siendo AnNsible el encargado de aplicar las configuraciones más o menos robustas según el entorno.

Los sistemas operativos que se van a usar para desplegar los distintos servicios son:

- Linux: Se usarán distintas distribuciones Linux para hacer un escenario más heterogéneo, estos Sistemas Operativos están basados en las principales distribuciones Linux y son los siguientes:
  - Debian 10
  - CentOS 7
- Windows: Se usaran Sistemas Operativo basados en Windows para los hosts de los usuarios y para dos servidores que tienen un Active Directory, estos son los siguientes:
  - Windows Server 2019
  - Windows Server 2012
  - Windows 10
  - Windows 7
  - Windows XP

A continuación podemos ver un diagrama simplificado de como será la arquitectura desplegada, los equipos que están dentro del cuadrado son aquellos que serán virtualizados.



{:height=
width="100%"}