

INGENIERÍA EN SISTEMAS COMPUTACIONALES.
SEGURIDAD Y VIRTUALIZACION.



PRACTICA 5: PROTECCIÓN CONTRA ATAQUES.

INTEGRANTES DEL EQUIPO:

JEANETTE ARLET SALAZAR NICOLÁS.	21620202
NELSY ORTIZ LÓPEZ.	21620165
MARIBEL LUCERO ZUÑIGA.	21620139

SEMESTRE: SEPTIMO

GRUPO: 7 US

ASESOR: EDWARD OSORIO SALINAS.

TLAXIACO, OAX, A 10 DE OCTUBRE DE 2024.

CONTENIDO

2.	ATAQUE DE FUERZA BRUTA	9
1.1	Definición.....	9
1.1	¿Qué ganan los hackers con los ataques de fuerza bruta?.....	9
1.2	Tipos de ataques de fuerza bruta.....	10
3.	ATAQUE DE DENEGACIÓN DE SERVICIO (DOS)	11
2.1	Definición.....	11
2.2	Como detectar un ataque DoS.....	11
4.	ATAQUE ECONOMICO DE DENEGACIÓN DE SERVICIO (EDoS)	12
3.1	Definición	12
5.	ATAQUE DE DENEGACIÓN DE SERVICIO DISTRIBUIDO (DDOS).....	12
4.1	Definición.....	12
4.2	¿Cómo funciona un ataque DDos?	13
6.	ATAQUE DE DENEGACIÓN DE SERVICIO POR AGOTAMIENTO DE RECURSOS.....	14
7.	ATAQUE DE DENEGACIÓN DE SERVICIO POR SATURACIÓN DE ANCHO DE BANDA.....	14
8.	BIBLIOGRAFÍA.....	15

TABLA DE ILUSTACIONES.

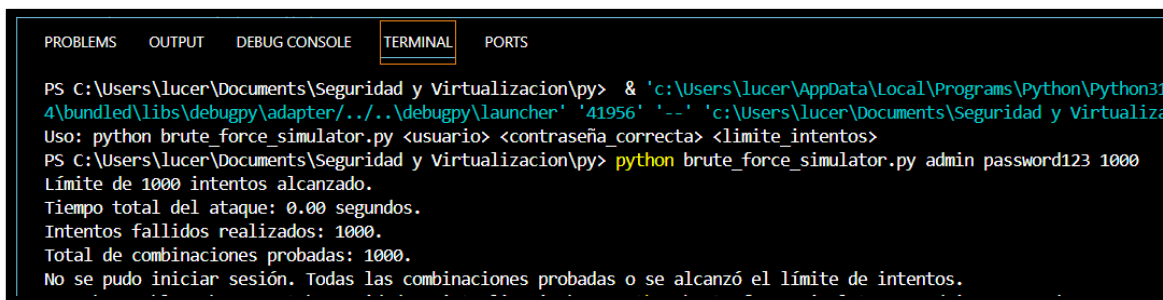
Ilustración 1:	Simulación de ataque de fuerza bruta, con un límite de 1000 intentos.....	3
Ilustración 2:	Simulación de ataque de fuerza bruta, con un límite de 1727604 intentos.	4
Ilustración 3:	Simulación de envío de 100 solicitudes a un servidor	4
Ilustración 5:	Ataque de fuerza Bruta.....	9
Ilustración 6:	Beneficios de los hackers al realizar ataques de fuerza bruta.....	9
Ilustración 7:	Ataque DDos.	13

DESARROLLO DE LOA PRACTICA.

1. Crea un programa que simule un ataque de fuerza bruta.

Este programa debe recibir un usuario y una contraseña, y debe intentar iniciar sesión en un sistema con estos datos. El programa debe intentar iniciar sesión con diferentes combinaciones de usuario y contraseña hasta que logre iniciar sesión o hasta que se alcance un límite de intentos fallidos.

- El programa debe recibir el usuario y la contraseña como argumentos de línea de comandos.
- El programa debe recibir el límite de intentos fallidos como argumento de línea de comandos.
- El programa debe mostrar un mensaje indicando si logró iniciar sesión o si se alcanzó el límite de intentos fallidos.
- El programa debe mostrar un mensaje indicando cuántos intentos fallidos se realizaron.
- El programa debe mostrar un mensaje indicando cuánto tiempo tardó en realizar el ataque.
- El programa debe mostrar un mensaje indicando cuántas combinaciones de usuario y contraseña se intentaron.



```
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> & 'c:\Users\lucer\AppData\Local\Programs\Python\Python314\
bundled\libs\debugpy\adapter\..\..\debugpy\launcher' '41956' '--' 'c:\Users\lucer\Documents\Seguridad y Virtualiz
Us: python brute_force_simulator.py <usuario> <contraseña_correcta> <limite_intentos>
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> python brute_force_simulator.py admin password123 1000
Límite de 1000 intentos alcanzado.
Tiempo total del ataque: 0.00 segundos.
Intentos fallidos realizados: 1000.
Total de combinaciones probadas: 1000.
No se pudo iniciar sesión. Todas las combinaciones probadas o se alcanzó el límite de intentos.
```

Ilustración 1: Simulación de ataque de fuerza bruta, con un límite de 1000 intentos

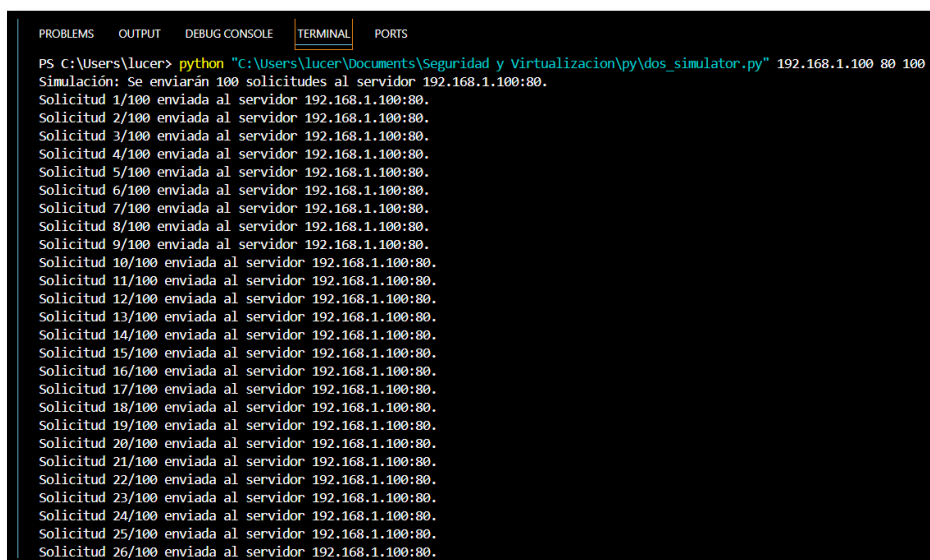
```
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> python brute_force_simulator.py admin password123 1727604
Límite de 1727604 intentos alcanzado.
Tiempo total del ataque: 0.44 segundos.
Intentos fallidos realizados: 1727604.
Total de combinaciones probadas: 1727604.
No se pudo iniciar sesión. Todas las combinaciones probadas o se alcanzó el límite de intentos.
```

Ilustración 2: Simulación de ataque de fuerza bruta, con un límite de 1727604 intentos.

2. Cree un programa que simule un ataque de denegación de servicio.

Este programa debe enviar una gran cantidad de solicitudes a un servidor para intentar saturarlo y evitar que responda a solicitudes legítimas.

- El programa debe recibir la dirección IP del servidor y el puerto como argumentos de línea de comandos.
- El programa debe recibir la cantidad de solicitudes a enviar como argumento de línea de comandos.
- El programa debe mostrar un mensaje indicando cuántas solicitudes se enviarán.
- El programa debe mostrar un mensaje indicando cuánto tiempo tardó en enviar las solicitudes.



```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\lucer> python "C:\Users\lucer\Documents\Seguridad y Virtualizacion\py\dos_simulator.py" 192.168.1.100 80 100
Simulación: Se enviarán 100 solicitudes al servidor 192.168.1.100:80.
Solicitud 1/100 enviada al servidor 192.168.1.100:80.
Solicitud 2/100 enviada al servidor 192.168.1.100:80.
Solicitud 3/100 enviada al servidor 192.168.1.100:80.
Solicitud 4/100 enviada al servidor 192.168.1.100:80.
Solicitud 5/100 enviada al servidor 192.168.1.100:80.
Solicitud 6/100 enviada al servidor 192.168.1.100:80.
Solicitud 7/100 enviada al servidor 192.168.1.100:80.
Solicitud 8/100 enviada al servidor 192.168.1.100:80.
Solicitud 9/100 enviada al servidor 192.168.1.100:80.
Solicitud 10/100 enviada al servidor 192.168.1.100:80.
Solicitud 11/100 enviada al servidor 192.168.1.100:80.
Solicitud 12/100 enviada al servidor 192.168.1.100:80.
Solicitud 13/100 enviada al servidor 192.168.1.100:80.
Solicitud 14/100 enviada al servidor 192.168.1.100:80.
Solicitud 15/100 enviada al servidor 192.168.1.100:80.
Solicitud 16/100 enviada al servidor 192.168.1.100:80.
Solicitud 17/100 enviada al servidor 192.168.1.100:80.
Solicitud 18/100 enviada al servidor 192.168.1.100:80.
Solicitud 19/100 enviada al servidor 192.168.1.100:80.
Solicitud 20/100 enviada al servidor 192.168.1.100:80.
Solicitud 21/100 enviada al servidor 192.168.1.100:80.
Solicitud 22/100 enviada al servidor 192.168.1.100:80.
Solicitud 23/100 enviada al servidor 192.168.1.100:80.
Solicitud 24/100 enviada al servidor 192.168.1.100:80.
Solicitud 25/100 enviada al servidor 192.168.1.100:80.
Solicitud 26/100 enviada al servidor 192.168.1.100:80.
```

Ilustración 3: Simulación de envío de 100 solicitudes a un servidor

```
Solicitud 94/100 enviada al servidor 192.168.1.100:80.  
Solicitud 95/100 enviada al servidor 192.168.1.100:80.  
Solicitud 95/100 enviada al servidor 192.168.1.100:80.  
Solicitud 96/100 enviada al servidor 192.168.1.100:80.  
Solicitud 97/100 enviada al servidor 192.168.1.100:80.  
Solicitud 98/100 enviada al servidor 192.168.1.100:80.  
Solicitud 99/100 enviada al servidor 192.168.1.100:80.  
Solicitud 100/100 enviada al servidor 192.168.1.100:80.  
Simulación completada. Se simularon 100 solicitudes en total.  
Tiempo total para simular las solicitudes: 0.02 segundos.
```

3. Documentar los resultados obtenidos y las posibles acciones que se pueden realizar para prevenir este tipo de ataques.

➤ RESULTADOS

Simulación de Ataque de Fuerza Bruta

Descripción:

- En esta simulación, el programa intenta iniciar sesión en un sistema utilizando diferentes combinaciones de contraseñas hasta que logra acceder o alcanza un límite de intentos fallidos.

Resultados:

- La simulación muestra cuántas combinaciones se intentaron, si se logró iniciar sesión, y cuánto tiempo tardó el ataque.
- Se puede observar cómo una combinación de nombre de usuario y contraseña débil puede ser susceptible a ataques de fuerza bruta.

Simulación de Ataque de Denegación de Servicio (DoS)

Descripción:

- En esta simulación, el programa envía un número específico de solicitudes a un servidor ficticio, midiendo el tiempo que toma enviar estas solicitudes.

Resultados:

- La simulación permite visualizar cómo se comportaría un servidor al recibir un gran número de solicitudes en un corto período de tiempo.
- El tiempo total para simular las solicitudes fue medido y reportado, junto con la cantidad de solicitudes enviadas.

➤ ACCIONES PREVENTIVAS

Modificaciones realizadas- Ataque Bruto

1. Control de intentos fallidos:

- Se agregó un contador `failed_attempts` que cuenta los intentos fallidos.
- Si se alcanzan 5 intentos fallidos, se muestra un mensaje indicando que la cuenta está temporalmente bloqueada y se finaliza el proceso.

2. Salida del Programa:

- Se mejoró el flujo de salida para que muestre el número de intentos fallidos y el total de combinaciones probadas.

Beneficios de las Modificaciones

- **Simulación de Prevención:**
 - Con estas modificaciones, se simula una protección contra ataques de fuerza bruta, ya que un atacante sería bloqueado después de múltiples intentos fallidos.
- **Conciencia de Seguridad:**

- El script ahora no solo simula un ataque, sino que también permite ver cómo se podrían implementar medidas para mitigar ataques similares en un entorno real.

```
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> & 'c:\Users\lucer\AppData\Local\Programs\24.10.0-win32-x64\bundled\libs\debugpy\adapter\..\..\debugpy\launcher' '44421' '--' 'c:\Users\lucer\Documents\Seguridad y Virtualizacion\py'
Uso: python brute_force_simulator.py <usuario> <contraseña_correcta> <limite_intentos>
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> python seguro.py admin password123 10
Demasiados intentos fallidos. La cuenta está temporalmente bloqueada.
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> █
```

Para Ataques de Denegación de Servicio (DoS):

Descripción de las mejoras

1. Limitación de Tasa:

- Se agregó un parámetro `rate_limit` que define cuántas solicitudes se pueden enviar en un intervalo de tiempo. Si se alcanza este límite, el programa espera un segundo antes de continuar con el envío de más solicitudes.

2. Esperanzas Aleatorias:

- Para simular un comportamiento más realista, se incluye una espera aleatoria entre 0,1 y 0,5 segundos entre cada solicitud, lo que puede ayudar a evitar detecciones de patrones.

3. Entrada del Puerto:

- Se asegura que el puerto se recibe como un entero.

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> & 'c:\Users\lucer\AppData\Local\Programs\Python\Python44491\python.exe' 'c:\Users\lucer\AppData\Local\Programs\Python\Python44491\Scripts\python.exe' 'c:\Users\lucer\Documents\Seguridad y Virtualizacion\py\debugpy\adapter\..\..\debugpy\launcher' '44491' '--' 'c:\Users\lucer\Documents\Seguridad y Virtualizacion\py\python Dos_seguro.py 192.168.1.100 80 100 10'
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py> python Dos_seguro.py 192.168.1.100 80 100 10
Simulación: Se enviarán 100 solicitudes al servidor 192.168.1.100:80.
Solicitud 1/100 enviada al servidor 192.168.1.100:80.
Solicitud 2/100 enviada al servidor 192.168.1.100:80.
Solicitud 3/100 enviada al servidor 192.168.1.100:80.
Solicitud 4/100 enviada al servidor 192.168.1.100:80.
Solicitud 5/100 enviada al servidor 192.168.1.100:80.
Solicitud 6/100 enviada al servidor 192.168.1.100:80.
Solicitud 7/100 enviada al servidor 192.168.1.100:80.
Solicitud 8/100 enviada al servidor 192.168.1.100:80.
Solicitud 9/100 enviada al servidor 192.168.1.100:80.
Solicitud 10/100 enviada al servidor 192.168.1.100:80.
Límite de tasa alcanzado: 10 solicitudes. Esperando para reanudar...
Solicitud 11/100 enviada al servidor 192.168.1.100:80.
Solicitud 12/100 enviada al servidor 192.168.1.100:80.
```

```
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS
Límite de tasa alcanzado: 10 solicitudes. Esperando para reanudar...
Solicitud 81/100 enviada al servidor 192.168.1.100:80.
Solicitud 82/100 enviada al servidor 192.168.1.100:80.
Solicitud 83/100 enviada al servidor 192.168.1.100:80.
Solicitud 84/100 enviada al servidor 192.168.1.100:80.
Solicitud 85/100 enviada al servidor 192.168.1.100:80.
Solicitud 86/100 enviada al servidor 192.168.1.100:80.
Solicitud 87/100 enviada al servidor 192.168.1.100:80.
Solicitud 88/100 enviada al servidor 192.168.1.100:80.
Solicitud 89/100 enviada al servidor 192.168.1.100:80.
Solicitud 90/100 enviada al servidor 192.168.1.100:80.
Límite de tasa alcanzado: 10 solicitudes. Esperando para reanudar...
Solicitud 91/100 enviada al servidor 192.168.1.100:80.
Solicitud 92/100 enviada al servidor 192.168.1.100:80.
Solicitud 93/100 enviada al servidor 192.168.1.100:80.
Solicitud 94/100 enviada al servidor 192.168.1.100:80.
Solicitud 95/100 enviada al servidor 192.168.1.100:80.
Solicitud 96/100 enviada al servidor 192.168.1.100:80.
Solicitud 97/100 enviada al servidor 192.168.1.100:80.
Solicitud 98/100 enviada al servidor 192.168.1.100:80.
Solicitud 99/100 enviada al servidor 192.168.1.100:80.
Solicitud 100/100 enviada al servidor 192.168.1.100:80.
Simulación completada. Se simularon 100 solicitudes en total.
Tiempo total para simular las solicitudes: 38.33 segundos.
PS C:\Users\lucer\Documents\Seguridad y Virtualizacion\py>
```

1. Investiga y describe los siguientes conceptos:

2. ATAQUE DE FUERZA BRUTA

1.1 Definición.

Método utilizado en ciberseguridad para obtener acceso no autorizado a sistemas, aplicaciones o datos. Este ataque consiste en probar sistemáticamente todas las combinaciones posibles de contraseñas o claves hasta encontrar la correcta. Los atacantes utilizan software automatizado



Ilustración 4: Ataque de fuerza Bruta.

que genera y prueba millones de combinaciones en un corto período de tiempo para lograr su objetivo. Es una técnica simple, pero efectiva, especialmente si las contraseñas son débiles o no cumplen con las mejores prácticas de seguridad (como tener una longitud adecuada o incluir caracteres especiales).

- proveedores de nube (AWS, Azure, Google Cloud, etc.), lo que les permitiría gestionar recursos virtuales a su favor.

1.1 ¿Qué ganan los hackers con los ataques de fuerza bruta?

Beneficios de los hackers	
Ganar dinero.	Ganar dinero: Pueden insertar anuncios spam en sitios web populares para ganar comisiones, redirigir tráfico a sitios pagados o instalar spyware para recopilar y vender datos de actividad a anunciantes.
Robo de datos personales.	Robo de datos personales: Pueden acceder a cuentas en línea para robar información sensible, como datos bancarios e información fiscal, o vender credenciales robadas. Incluso pueden filtrar bases de datos completas de organizaciones.
Difusión de malware.	Difusión de malware: Redirigen el tráfico a sitios maliciosos o infectan directamente un sitio web para propagar malware en los dispositivos de los visitantes.
Secuestro de sistemas.	
Arruinar la reputación de sitios web.	

Ilustración 5: Beneficios de los hackers al realizar ataques de fuerza bruta.

Secuestro de sistemas: Utilizan malware para convertir dispositivos en parte de una **botnet** y realizar actividades maliciosas a gran escala, como ataques DDoS.

Arruinar la reputación de sitios web: Pueden sabotear sitios web con contenido ofensivo o inapropiado, dañando su imagen y credibilidad.

1.2 Tipos de ataques de fuerza bruta

Cada ataque de fuerza bruta puede utilizar diferentes métodos para descubrir tus datos sensibles. Podrías estar expuesto a cualquiera de los siguientes métodos populares de fuerza bruta:

Ataques simples de fuerza bruta: Los hackers intentan adivinar lógicamente tus credenciales, sin ayuda de herramientas de software u otros medios. Estos pueden revelar contraseñas y PIN sencillos. Ejemplo, una contraseña como **“guest12345”**.

Ataques de diccionario

El hacker elige un objetivo y comprueba las posibles contraseñas con ese nombre de usuario. Los ataques de diccionario son la herramienta más básica en los ataques de fuerza bruta. Algunos hackers utilizan diccionarios íntegros y amplían palabras con ayuda de caracteres especiales y números, o bien utilizan diccionarios especiales, aunque este método de ataque secuencial resulta engorroso.

Ataques híbridos de fuerza bruta: Estos hackers mezclan medios externos con sus conjeturas lógicas para intentar una intrusión, suele mezclar ataques de diccionario y de fuerza bruta. Estos ataques se utilizan para averiguar contraseñas combinadas que mezclan palabras comunes con caracteres aleatorios.

Ataques de fuerza bruta inversos: Invierte la estrategia de ataque comenzando con una contraseña conocida. A continuación, los hackers buscan en millones de nombres de usuario hasta encontrar una coincidencia. Muchos

de estos delincuentes empiezan con contraseñas filtradas que están disponibles en Internet a partir de filtraciones de datos ya existentes.

Relleno de credenciales: Si un hacker tiene una combinación de nombre de usuario y contraseña que funciona en un sitio web, la probará también en muchos otros. Dado que se sabe que los usuarios reutilizan la información de inicio de sesión en muchos sitios web, son el objetivo exclusivo de un ataque de este tipo.

3. ATAQUE DE DENEGACIÓN DE SERVICIO (DOS)

2.1 Definición.

Tipo de ciberataque en el que un actor malicioso tiene como objetivo que un ordenador u otro dispositivo no esté disponible para los usuarios a los que va dirigido, interrumpiendo el funcionamiento normal del mismo. Los ataques DoS suelen funcionar al sobrecargar o inundar una máquina objetivo con solicitudes hasta que el tráfico normal es incapaz de ser procesado, lo que provoca una denegación de servicio a los usuarios de la adición. Un ataque DoS se caracteriza por utilizar un único ordenador para lanzar el ataque.

2.2 Como detectar un ataque DoS.

Para detectar si estás siendo víctima de un ataque de denegación de servicio (DoS), hay varios signos y métodos que pueden ayudarte a identificarlo:

Baja en el Rendimiento del Sistema: Puede hacer que el rendimiento de tu sistema o red disminuya drásticamente. Las páginas web pueden cargar más lentamente o no responder.

Aumento Súbito del Tráfico de Red: Aumento repentino e inusual en el tráfico de red. Este tráfico puede parecer provenir de una sola fuente o de una serie de direcciones IP similares.

Conexiones y Sesiones Saturadas: Puede llenar las tablas de sesiones y conexiones de tu sistema. Esto se puede monitorear usando herramientas que analizan la cantidad de conexiones activas en tu servidor.

Monitoreo de Recursos: Utiliza herramientas de monitoreo de rendimiento para revisar el uso de la CPU, memoria y ancho de banda. Un ataque DoS a menudo consume estos recursos rápidamente. Un aumento inusual o sostenido en el uso de recursos podría indicar un ataque en curso.

Registros del Sistema (Logs): Revisar los registros del sistema (logs) para buscar patrones inusuales, como múltiples solicitudes del mismo origen en un corto período de tiempo o intentos de conexión fallidos.

4. ATAQUE ECONOMICO DE DENEGACIÓN DE SERVICIO (EDoS)

3.1 Definición

Es una variante del DoS que no solo busca interrumpir el servicio, sino que también se enfoca en causar un impacto económico significativo a la víctima. En entornos de virtualización y servicios en la nube, donde los recursos se escalan automáticamente bajo demanda, un EDoS intenta saturar el sistema para que la víctima incurra en costos elevados al tener que provisionar más recursos para manejar la carga artificial del ataque.

5. ATAQUE DE DENEGACIÓN DE SERVICIO DISTRIBUIDO (DDoS).

4.1 Definición.

Similar a un DoS, un DDoS también tiene como objetivo saturar un sistema para que no pueda responder a los usuarios legítimos, pero en lugar de usar un solo dispositivo para lanzar el ataque, se utilizan múltiples dispositivos distribuidos en diversas ubicaciones, llamados botnets. Esto hace que el ataque sea más potente y difícil de mitigar. En un entorno virtualizado, un DDoS puede colapsar varios servidores virtuales simultáneamente, afectando la estabilidad y disponibilidad de servicios en la nube. Los ordenadores que realizan el ataque DDoS son reclutados mediante la infección de un malware, convirtiéndose así en bots o zombis, capaces

de ser controlados de forma remota por un ciberdelincuente. Un conjunto de bots, es decir, de ordenadores infectados por el mismo malware, forman una botnet o también conocida como red zombi. Obviamente, esta red tiene mayor capacidad para derribar servidores que un ataque realizado por sólo una máquina.

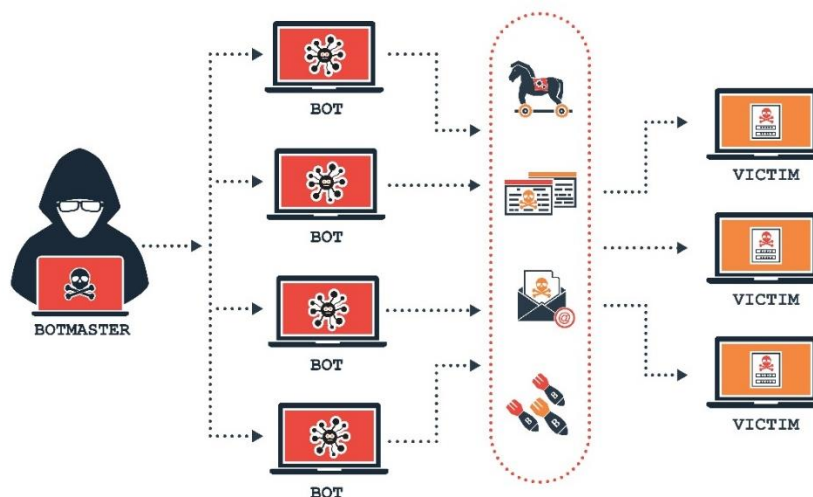


Ilustración 6: Ataque DDoS.

4.2 ¿Cómo funciona un ataque DDoS?

Los atacantes utilizan malware para crear una red de bots: dispositivos conectados a Internet que están infectados con malware y que los atacantes pueden dirigir para enviar una gran cantidad de tráfico a los objetivos. Esta red de bots, o botnet, puede incluir terminales como el Internet de las cosas (dispositivos de IoT), smartphones y ordenadores personales, así como enrutadores y servidores de red. Cada dispositivo infectado es capaz de propagar el malware a otros dispositivos para amplificar las dimensiones de un ataque.

Lanzamiento de un ataque: Una vez que un atacante ha creado una botnet, envía instrucciones remotas a los bots, indicándoles que envíen solicitudes y tráfico a un servidor, sitio web, aplicación web, API o recurso de red objetivo. Esto crea una cantidad de tráfico que provoca una denegación de servicio, impidiendo que el tráfico normal acceda al objetivo.

DDoS como servicio: A veces, las botnets, con sus redes de dispositivos afectados, se alquilan para otros posibles ataques a través de servicios de "ataques de alquiler". Esto permite que cualquiera que no tenga buenas intenciones y carezca de formación o experiencia pueda perpetrar ataques DDoS fácilmente.

6. ATAQUE DE DENEGACIÓN DE SERVICIO POR AGOTAMIENTO DE RECURSOS

Este tipo de ataque se enfoca en consumir los recursos del sistema objetivo, como memoria, CPU o conexiones de red, hasta que el servicio se vuelva lento o deje de funcionar. En un entorno de virtualización, estos ataques pueden agotar los recursos de las máquinas virtuales, afectando las aplicaciones y servicios que dependen de ellas.

7. ATAQUE DE DENEGACIÓN DE SERVICIO POR SATURACIÓN DE ANCHO DE BANDA

En este tipo de ataque, el objetivo es saturar la conexión de red del sistema enviando grandes volúmenes de datos, sobrepasando la capacidad del ancho de banda disponible. En un entorno virtualizado, esto puede causar que las máquinas virtuales y los servicios alojados en ellas se vuelvan inalcanzables, afectando el acceso y la disponibilidad de las aplicaciones y servicios en línea.

8. BIBLIOGRAFÍA

- akamai. (s.f.). Retrieved 08 de octubre de 2024, from <https://www.akamai.com/es/glossary/what-is-ddos>
- *cloudflare*. (s.f.). Retrieved 08 de octubre de 2024, from <https://www.cloudflare.com/es-es/learning/ddos/glossary/denial-of-service/>
- kaspersky. (s.f.). *kaspersky*. <https://www.kaspersky.es/resource-center/definitions/brute-force-attack>