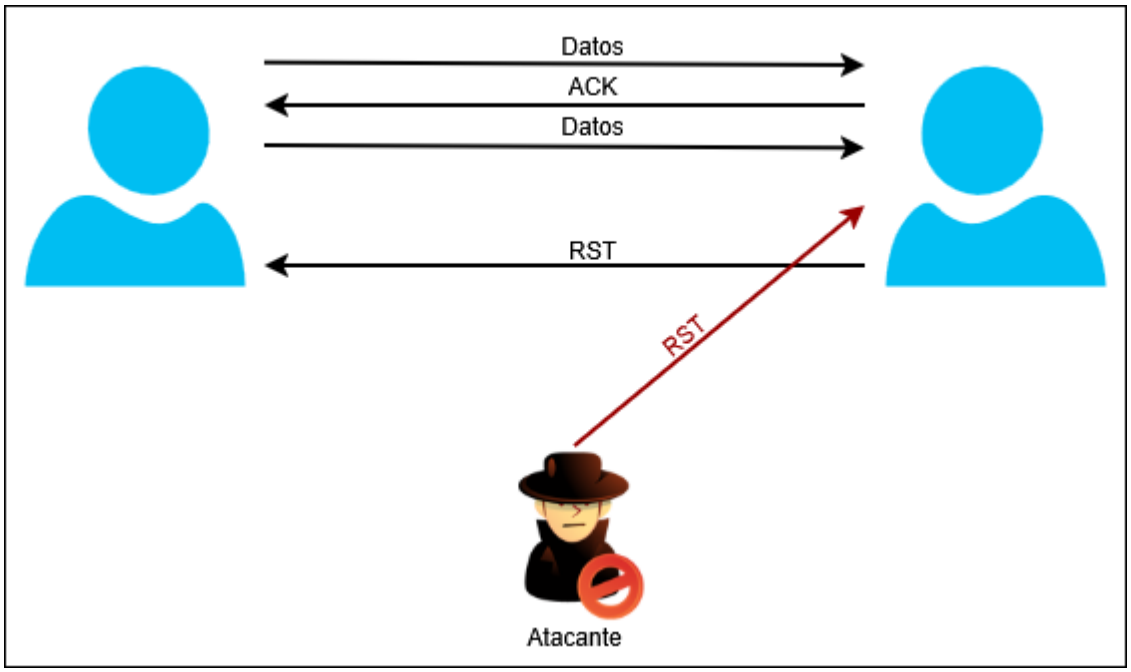
Securizando la infraestrucura de internet

Problemas de seguridad comunes en TCP

Disponibilidad

Ataque TCP reset

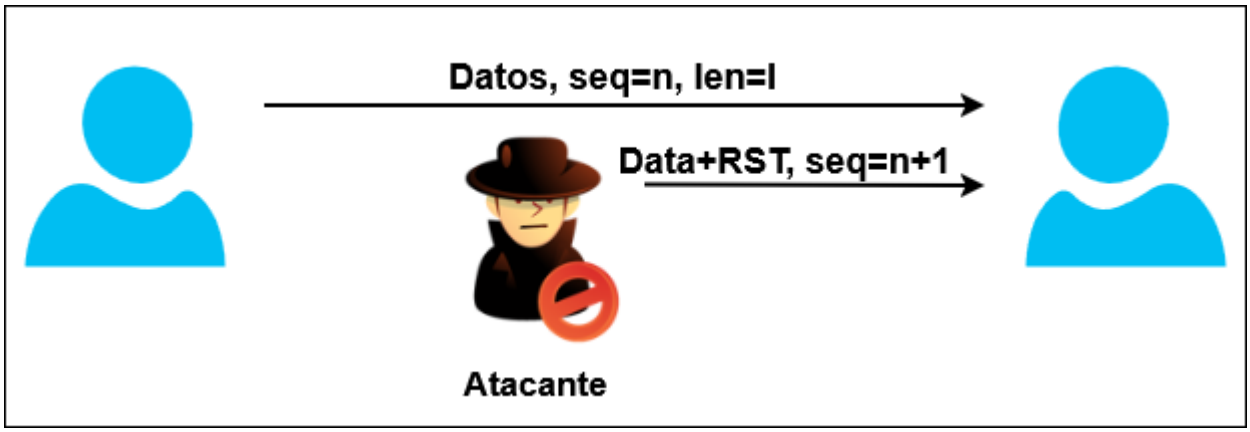
El Flag ReSeT (RST) provoca la caida de la conexión, normalmente se usa para recuperación de errores. El atacnte ijecta un marco con el flag RST activo, provocando que el receptor corte la conexión inmediatamente.



El ataque tiene los siguientes requisitos:

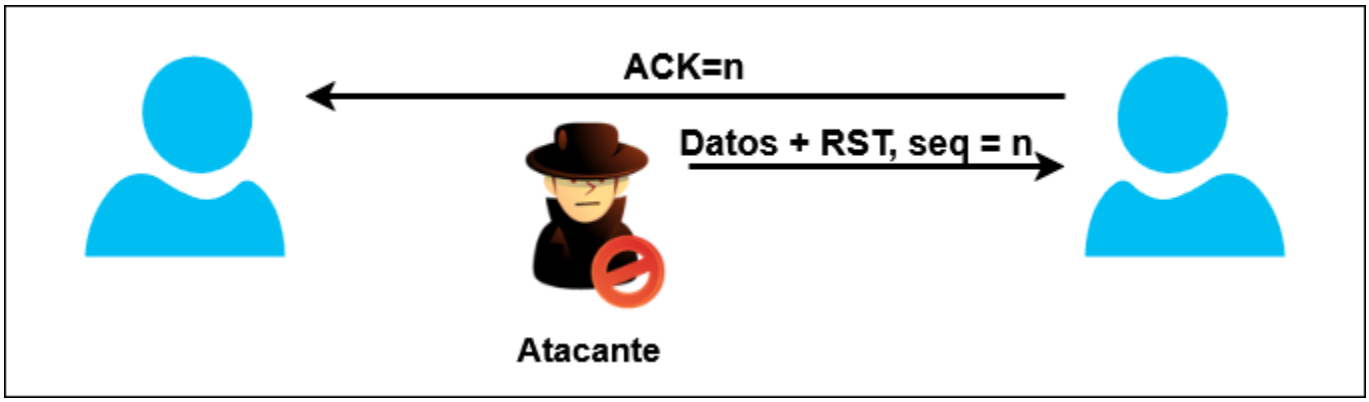
- RST se encuentra dentro del rango permitido
  - En todos los estados salvo SYN-SENT, todos los fragmentos RST son validados revisando sus campos SEQ (Número de secuencia).
  - Un reset es válido si el número de secuencia está dentro del rango
  - En el estado SYN-SENT el RST es aceptable si el campo ACK reconoce el SYN
  - Número de escuencia entre RCV.NXT y RCV.NXT+RCV.WND
  - Rangos históricos < 64 kbytes.
- Right 4-tuple
  - Se deben conocer la IP y puerto del server
  - Se deben conoce la Ip y el puerto del cliente

Ataque TCP reset - Sin limitación de posición: Atacante en la red del cliente



- El atacante inspecciona los datos enviados por el primer equipo al segundo equipo
- Antes de que el primer equipo envíe el siguiente paquete, el atacante genera un paquete RST válido con:
  - Misma Dirección IP
  - Mismos Puertos TCP
  - Número de secuencia correcto.

Ataque TCP reset - Sin limitación de posición: Atacante en la red del servidor



- El atacante inspecciona el tráfico del segundo equipo al primero
- Antes de que el primer equipo envíe el siguiente paquete, se genera un paquete RST válido
  - Misma IP
  - Mismos puertos TCP
  - Número de secuencia correcto

Ataque TCP reset - Posición limitada: Blind Data/RST Injection

Es un ataque difícil de realizar, las direcciones de ambos extremos no suelen ser conocidas, aunque se suelen conocer las direcciones de los servidores, las de los servidores suelen ser desconocidas. Los puerto de los dos extremos suelen ser desconocidos, el de server a veces es conocido, pero el de los clientes suele ser impredecible. El número de secuencia tampoco es conocido. Las conexiones suelen tener un tiempo de vida muy corto y el rango del número de secuencia valido cambia, por lo que cualquier intento de adivinar estos datos es poco útil.

Por otro lado, los protocolos tienen un amplio tiempo de vida y las direcciones se pueden saber por adelantado.

Defensas contra ataque TCP reset

- Solo aceptar el segmento RST si el número de secuencia es el primero en el rango (proveido por el sistema operativo)
- Filtrar paquetes spoofeados a nivel IP (Tiene que ser realizado por los servidores de autenticación en los extremos)
- Usar tmarcas de tiempo como defensa adicional: PAWS (Protection Against Wrapped Sequences)
- Paquetes TCP autenticados (TCP-AO)

Ataque SYN Flooding

Una tacante agota la tabla de conexiones

1. El atacante envía paquetes SYN con su dirección real pero nunca completa las conexiones (Filtrado fácilmente del lado del servidor)
2. El atacante envía paquetes SYN con direcciones suplantadas (El receptor suplantado envía RST y el servidor elimina la conexión)
3. El atacante envía paquetes SYN con direcciones suplantadas que no responden (Ataque DoS)

Para defendernos de SYN Flood podemos hacer lo siguiente:

- Dropear conexiones
- Reducir el uso de memoria para conexiones noe stablecidas
- No almacenar nada (Como se establece la conexión entonces?)

SYN Cookies

Se cifra la información de conexión dentro del ISN

- Primeros 5 bits: número lentamente incremental basado en el tiempo
- Siguientes 3 bits: Se codifica el MSS anunciado del cliente
- Últimos 24 bits: Hash secreto de la IP y portes del cliente y los primeros 5 bits.

Cuando se recibe un ACK sin una conexión establecida:

1. Se sustrae 1 del número ACK
2. Se Recalcula el hash y se compara con los últimos 24 bits, si el hash coincide, se almacena la ifnormación de la conexión.

Este mecanismo no necesita cooperación por parte del cliente.

Las SYN Cookies tienen algunos puntos negativos:

- Falta de espacio para negociar opciones

    - MSS: solo 3 bits, aproximadamente 65000 valores
    - SACK: No hay espacio para el
    - Rangos grandes: No hay espacion para ellos
- Soluciones:
    - Incrementar los bits para codificación de información (Reduce la resistencia del hash)
    - Usar SYN Cookies solo cuando se está bajo ataque, mejor tener mal rendimiento que perder el servicio
    - Tomar ventaja de otras opciones para incrementar el espacio
        - Marca de tiempo del servidor enviada de vuelta en CAK, usa la parte final para cofigicar información

---

Ataque SlowLoris sobre HTTP

Busca agotar la thread pool de un servidor web:

1. Se hacen muchas peticiones HTTP incompletas
2. Se envían encabezados periódicamente para mantenerlos abiertos
3. Nunca cerrar y sie el server cierra, volver a abrir

Esto se puede mitigar de la siguiente manera:

- Incrementar el número de hilos (Poco efectivo)
- Limitar Tiempo de conexión, conexiones por IP…
- Proxy inverso en la nube: No se envía nada al servidor web.

Este ataque también se puede realicar sobre QUIC

Autenticación

La ausencia de autenticación tiene los siguientes problemas:

- Identificación de conexión
    - Puerto e IP de destino: Conocidos
    - Puerto de origen: puede ser adivinado
    - IP de origen: Puede ser suplantada
- La falta de autenticación puede llevar a:
    - Ataques DoS: RST attack y SYN flooding
    - Ataques de inyección de datos.

Autenticación basada en posición

Controla la conexión entre host y routers adyacentes donde ambas partes residen en la misma LAN o se sabe que ambas partes están como mucho a n saltos de distancia.

Generalized TTL Security Mechanism (GTSM)

El procedimiento de transmisión consiste en enviar todos los datagramas IP con TTL/Hop limt de 255, de forma que podemos saber que los paquetes solo han dado n saltos. Los datagramas relacionados con ICMP también usan TTL=255. En recepción:

- Unknown: Cualquier datagrama IP no telacionado con una sesión protegida GTSM
- Trusted: Un datagrama de una sesión GTSM con valor TTL correcto, normalmente 254
- Dangerous: Un datagrama de una sesión GTSM con valor TTL incorrecto. Se les da poca prioridad o se dropean para evitar DoS.

TCP Authentication Option (TCP-AO)

Tiene como objetivo proteger la capa de transporte para conexiones de protocolo de enrutado de larga duración y cualquier otra conexión de larga duración, sustituye TCP-MD5. Complementa IPsec e IKE. TCP-AO esta pensada para IPSec inviable, protegiendo los protocolos de enrutado. Mientras que TLS protege los datos, TCP-AO protege los protocolos de información.

TCP-AO es una mejora sobre TCP-MD5 ya que provee de algoritmos más fuertes, seguridad Two-Fold (Klaves de tráfico generadas a partir de la clave configurada por el usuario), mejor manejo de claves y agilidad (Cambio de claves al momento sincronizando el cambio entre los dos lados de la conexión) y es mejor para conexiones de larga duración.

Claves TPC-AO

- Master Key Tuples (MKT): Describe propiedades asociadas a las conexiones
  - ID: número único representando un MKT
  - Identificador de Conexión TCP: Direcciones ip y puertos asociados con la conexión
  - TCP Option Flag: Opciones para ser autenticado
  - Master Key: Secuencia de bits aleatoria segura usada para generar las claves de tráfico (Traffic keys)
  - Key Derivation Function (KDF)
  - Algoritmo MAC
- Traffic Keys: Derivadas de MKT, ambas direcciones IP, ambos puertos y ambos ISN
- Send_SYN_traffic_key: No usa ISN
- Receive_SYN_traffic_key: No usa ISN, solo para conexiones abiertas simultáneamente
- Send_other_traffic_key
- Receive_other_traffic_key

Protegiendo el DNS

From:
https://knoppia.net/ - **Knoppia**

Permanent link:
https://knoppia.net/doku.php?id=master_cs:secom:tm4_v2&rev=1779977023

Last update: **2026/05/28 14:03**