## **Key Findings**

### **1. Attack Identification**

* **Type of Attack:** **SYN Flood (Denial-of-Service Attack)**
* **Malicious Source IP:** **203.0.113.0**
* **Target Server:** **192.0.2.1 (HTTPS Port 443)**
* **Attack Pattern:** Rapid, repeated **SYN packets** without completing the TCP handshake.

### **2. Attack Mechanism**

* The attacker sent **hundreds of SYN packets** in quick succession.
* The server responded with **SYN-ACK**, but the attacker **never sent the final ACK**, leaving connections half-open.
* Legitimate users (**198.51.100.x**) experienced **failed connections (RST, ACK)** and **timeouts (504 Gateway Timeout)** due to resource exhaustion.

### **4. Impact**

* **Server Overload:** The server was unable to handle legitimate requests due to **half-open connections**.
* **Service Disruption:** Legitimate users received **504 Gateway Timeout** errors.
* **Resource Exhaustion:** The attack consumed server resources, leading to **connection resets (RST, ACK)** for valid traffic.

## **Mitigation Recommendations**

### **Short-Term Actions**

1. **Block the Attacker’s IP (203.0.113.0)** at the firewall level.
2. **Enable SYN Cookies** to mitigate SYN flood attacks.
3. **Rate-Limit SYN Packets** to prevent overwhelming the server.

### **Long-Term Solutions**

1. **Deploy a DDoS Protection Service** (e.g., Cloudflare, AWS Shield).
2. **Implement Intrusion Detection/Prevention Systems (IDS/IPS)** to detect and block malicious traffic.
3. **Monitor Network Traffic** for unusual SYN packet patterns.

## **Conclusion**

The logs confirm a **SYN flood attack** from **203.0.113.0**, disrupting legitimate traffic. Immediate mitigation is required to restore service availability. Future prevention should include **DDoS protection** and **traffic filtering**.

**Next Steps:**

* Apply firewall rules to block the attacker.
* Enable SYN flood protection mechanisms.
* Review server capacity and scaling options.