**Cyber Lab Ex1 – SYN Flood**

הסבר כללי: **במטלה זו קיבלנו Docker שמכיל 3 מכונות – Server, Attacker, Monitor.**

**התבקשנו לבצע התקפת SYN Flood באמצעות מכונת ה-Attacker על ה-Server.**

**רעיון ההתקפה:**

**מכונה אחת (Attacker) שולח הודעות SYN רבות לשרת (Server).**

**השרת מחזיר הודעת SYN-ACK לכל הודעה כזו, ומקצה משאבים ע"מ להמשיך תקשורת עם המכונה.**

**מכיוון שכמות הודעות ה-SYN גדולה – השרת מקצה משאבים רבים לצורך התקשורת, מה שמוביל למצב בו הוא לא קיימים לא משאבים מספיקים בשביל לתקשר עם מכונות אחרות. וכך בעצם אנו מבצעים Denial Of Service לשרת.**

**בפועל מבוצע Distributed Denial Of Service, שכן אנו מזייפים כתובות IP כדי שהשרת לא ידע איזה IP לחסום. וכך בעצם אנו מונעים מהשרת לעצור את ההתקפה.**

**מטרת העל של כל Image:**

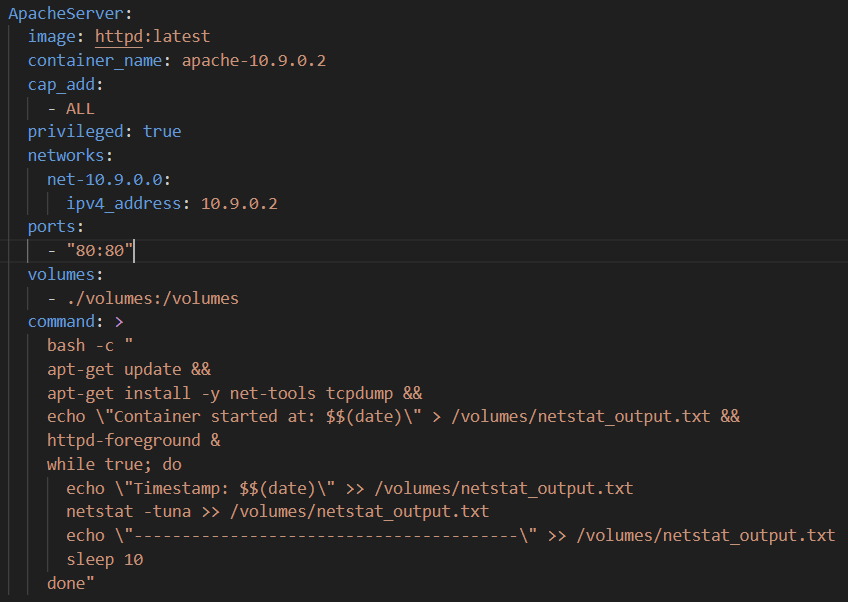
1. **Server – שרת Apache2. משמש בתור מטרה לתוקף.**
2. **Attacker – לבצע שליחה של חבילות SYN מרובות על גבי פרוטוקול TCP לפורט 80**

**ב-Server.**

1. **Monitor – שליחת PING לשרת ע"מ לנתר את העומס על השרת בזמן המתקפה. זאת ע"י בדיקת זמני RTT לפני ההתקפה, במהלך ההתקפה ואחריה.**

**הסבר על כל Image:**

**Server**

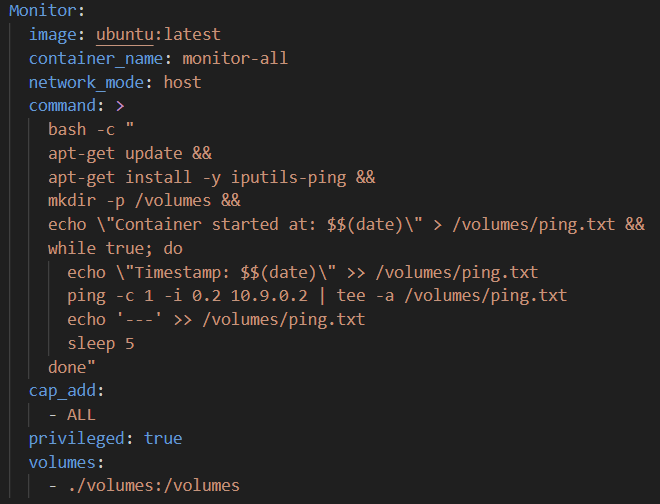
****

**שרת HTTP בכתובת 10.9.0.2 שמבצע חלק פסיבי בהתקפה בכך שהוא המטרה של התוקף.**

**כאשר ה-Image של השרת עולה. השרת מתקין כלי שנקרא tcpdump, מה שמאפשר לנו לנתר את כמות החיבורים במצב SYN RECV (חיבורים מהם השרת קיבל הודעת SYN וממתין להודעת ACK על הודעת ה-SYN ACK אותה שלח).**

**אנו מדפיסים לקובץ את תוצאת netstat -tuna כל 10 שניות. כך אנו מתעדים את השינוי בחיבורים הפתוחים בשרת בין זמן ההתקפה לזמן שלפניה ואחריה.**

**Monitor**

****

**מכונת ה-Monitor שולחת הודעת Ping**