Malicious Honeypot Test Results

Overall Test Plan

Development of the project consisted of setting up a server to host the Honeypot and then implementing features like the logger, and email alerts.

Test Case Descriptions and Results

Server Connectivity Test 1 (SCT1)

- This test will determine if we can connect to the server when on the same local network
- Result: Success
    - SSH connection was achieved with the server while on the LAN

Server Connectivity Test 2 (SCT2)

- This test will determine if we can connect to the server remotely from a foreign network
- Result: Success
    - SSH connection was achieved via an Android SSH client running on a seperate network.

Server Connectivity Test 3 (SCT3)

- This test will determine if users other than ourselves can find the server and attempt to log into it
- Result: Success
    - SSH attempts were documented in /var/log/auth.log on the server after forwarding port 22

Alert Test 1 (AT1)

- This test will determine if we can alert the admin to show that activity is taking place on the honeypot
- Result: Success
    - Email was sent after a successful login occured on the server.

Alert Test 2 (AT2)

- This test will determine if we can alert the admin to show that an attacker has gained access to the honeypot
- Result: Untested
    - Seemed like a lower priority given the number of group members that knew about the Honeypot so this feature has not been implemented yet.

Logger Test 1 (LT1)

- This test will determine if we can record all activity on the server to a log file
- Result: Partial success
    - There were already some log files that the server created on its own that we could read from and modify as needed

Logger Test 2 (LT2)

- This test will determine if we can trigger the logger to record all activity on the server to a log file
- Result: Failure

    - Attempted to write a script to the /etc/profile to record all commands of a user that was triggered by the log in. Bricked the server because it was a read only file, and bricked a VM trying to test another attempt. Repairing the server/VM was time consuming so this was left behind.

Packet Dump Test 1 (PDT1)

- This test will determine if we can retrieve packets from server sessions
- Result: Success
    - Were able to retrieve specified number of packets from server sessions

Packet Dump Test 2 (PDT2)

- This test will see if we can write the results of the packet dump into a log file
- Result: Untested

Packet Dump Test 3 (PDT3)

- This test will see if we can make the alert system trigger the packet dump
- Result: Untested
    - Lower priority feature to be implemented

Packet Dump Test 4 (PDT4)

- This test will see if we can stop the packet dump after a server session has ended
- Result: Untested
    - Lower priority feature to be implemented

Spoof Command Test 1 (SpCT1)

- This test will determine if commonly run linux commands can be masqueraded
- Result: Untested
    - Had a fair bit of trouble creating simpler scripts in Linux that bricked the server. Did not want to risk spending too much time on an advanced concept and brick the server again.

Malicious Access Test 1 (MAT1)

- This test will see if we can gain access to the attacker’s system with information retrieved from the server session
- Result: Untested
    - Had some information to use, but ultimately were unable to coordinate an counter attack by exploiting vulnerable services