Ex0c0

Joshua Main-Smith

2020-10-27

Contents

Technical Report

Finding: Local Users Can Connect via RDP

Risk Assessment

Any user within the local area network can connect to Patronum via RDP. This would allow an attacker to gain remote access provided the proper credentials are readily available.

Vulnerability Description

The MS RDP port on Patronum is open and listening for connections. Any user that has access to the internal network can connect to the Patronum machine via RDP.

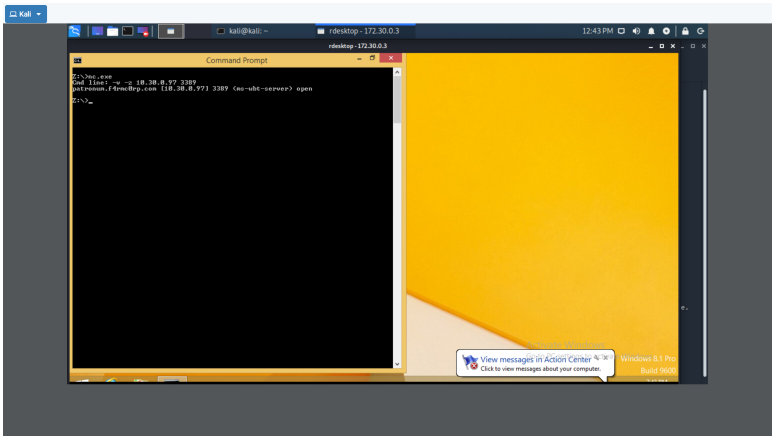
Mitigation or Resolution Strategy

Unless MS RDP is absolutely necessary, port 3389 (MS RDP) should be closed to any incoming connections.

Attack Narrative

Scanning Patronum

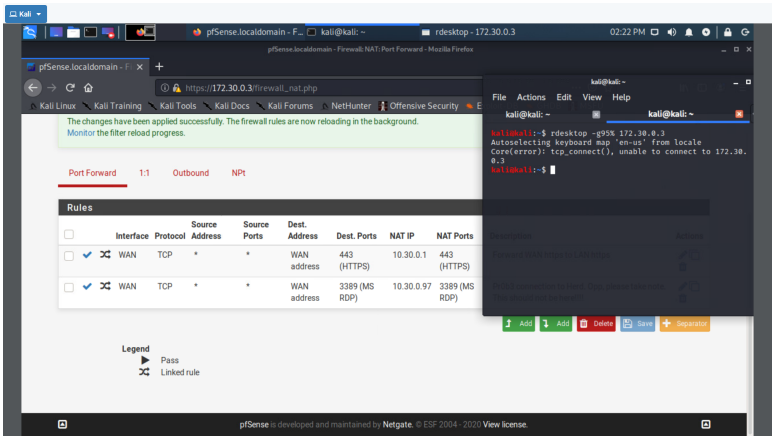
To start, we use **rdesktop** to access Brian's account on Herd like we've done in the previous exercise (the port forward to Herd is already set up on PfSense). We included the netcat binary under **/usr/share/windows-resources/binaries/nc.exe** to scan for open ports on Patronum under Herd. Once we connected to Herd, we scanned for port 3389 using netcat.exe to see if MS RDP is open on Patronum. To do this, we started netcat with **nc.exe** then issued the command **-v -z 10.30.0.97 3389**. The result was an open port, as shown below.



This indicates that we can RDP into Patronum from Herd.

Attempting a Direct Patronum Connection

Since MS RDP is open on Patronum it would be easier to connect directly to Patronum from our Kali host using port forwarding on PfSense, like we have been doing with Herd. So, we changed the port forwarding configuration on PfSense to forward all RDP connections to Patronum (10.30.0.97). While attempting to RDP into Patronum from our Kali host, we received an error that we were unable to connect.



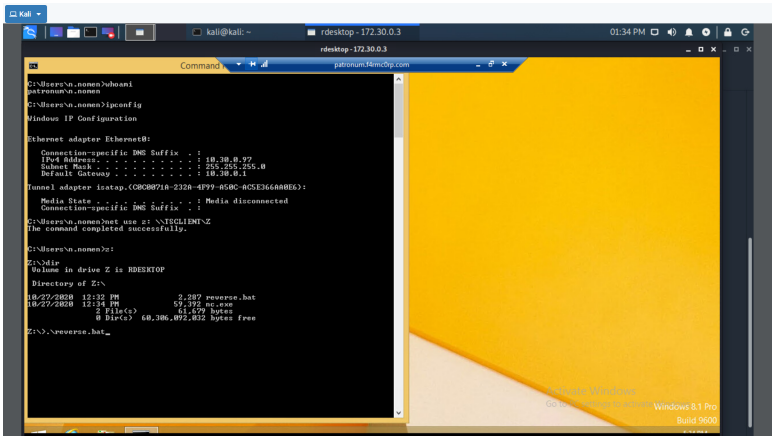
This is most likely due to Patronum accepting local area connections and rejecting wide area connections, unlike Herd accepting both local area and wide area connections. So, to proceed we needed to use Herd as a pivot in connecting to Patronum.

Connecting to Patronum

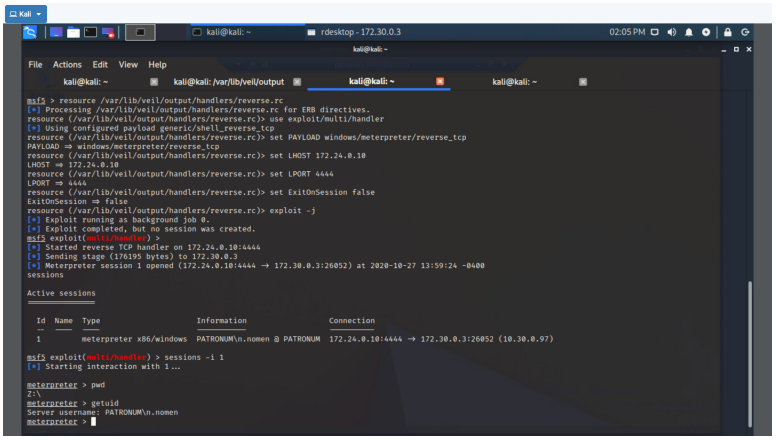
Before moving onto Patronum, we need to create reverse_tcp batch script using veil so that we can avoid antivirus detection on the target host. To do this, we ran **veil** in the terminal. We used the **Evasion** setup by selecting **use 1**. Listing all the payloads, we decided to use the **powershell/meterpreter/rev_tcp** option with **use 22**. The only thing we needed to do from here is set the listening host to our machine's IP (172.24.0.10).

We can then **generate** the files, which creates a resource file used by Metasploit to listen for any incoming connections, and a batch files for running on Patronum. The batch file created from veil is located under **/var/lib/veil/output/sources/batchfile.bat**, which we include while connecting to Herd via RDP. We also need to set up a listener on Metasploit by issuing the msf command **resource /var/lib/veil/output/handlers/resource.rc**, which will start a reverse tcp handler on port 4444.

On Herd, we can start MS RDP by using the command **mstsc**. The computer we want to connect to is **patronum.f4rmc0rp.com** and we include our drive containing our batch script that we mounted on Herd under **Local Resources**. We are able to successfully connect under *n.nomen* using the credentials we found hash cracking the logins under Herd.



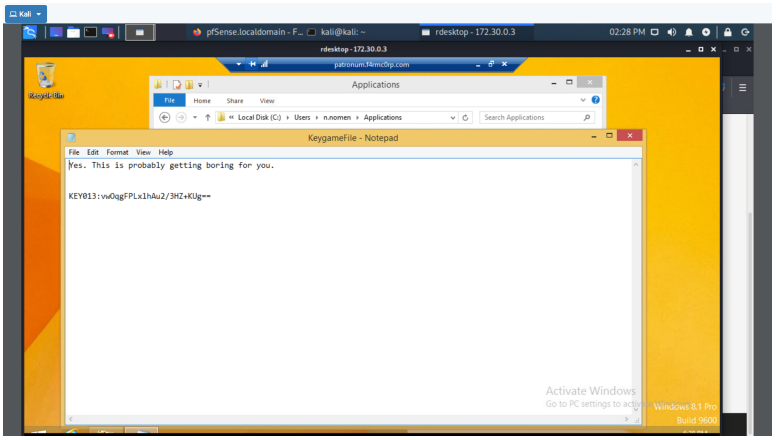
Once we gained access, we confirmed that we successfully logged in as N. Nomen on Patronum, as can be seen above. We can then mount our folder containing the batch file, then run **batchfile.bat** so that we can communicate directly from our attack box. This is confirmed in the screenshot below.



We were able to successfully view and exfiltrate files directly from Patronum to our attack box.

KEY

While exploring the file system of Patronum, we found a key under N. Nomen's Applications folder. The key has the value: **KEY013:vw0qgFPLxlhAu2/3HZ+KUg==**.



Zoom Link

October 26, 2020