Threat Intelligence

Bypassing Network Restrictions Through RDP Tunneling

January 24, 2019

Mandiant

Written by: David Pany, Steve Miller, Danielle Desfosses

Remote Desktop Services is a component of Microsoft Windows that is used by various companies for the convenience it offers systems administrators, engineers and remote employees. On the other hand, Remote Desktop Services, and specifically the Remote Desktop Protocol (RDP), offers this same convenience to remote threat actors during targeted system compromises. When sophisticated threat actors establish a foothold and acquire ample logon credentials, they may switch from backdoors to using direct RDP sessions for remote access. When malware is removed from the equation, intrusions become increasingly difficult to detect.

RDPing Against the Rules

which can leave unwanted artifacts on a system. As a result, FireEye has observed threat actors using native Windows RDP utilities to connect laterally across systems in compromised environments. Historically, non-exposed systems protected by a firewall and NAT rules were generally considered not to be vulnerable to inbound RDP attempts; however, threat actors have increasingly started to subvert these enterprise controls with the use of network tunneling and host-based port forwarding.

Network tunneling and port forwarding take advantage of firewall "pinholes" (ports not protected by the firewall that allow an application access to a service on a host in the network protected by the firewall) to establish a connection with a remote server blocked by a firewall. Once a connection has been established to the remote server through the firewall, the connection can be used as a transport mechanism to send or "tunnel" local listening services (located inside the firewall) through the firewall, making them accessible to the remote server (located outside the firewall), as shown in Figure 1.

Figure 1: Enterprise firewall bypass using RDP and network tunneling with SSH as an example

Inbound RDP Tunneling

establish secure shell (SSH) network connections to other systems using arbitrary source and destination ports. Since many IT environments either do not perform protocol inspection or do not block SSH communications outbound from their network, attackers such as FIN8 have used Plink to create encrypted tunnels that allow RDP ports on infected systems to communicate back to the attacker command and control (C2) server.

```
Example Plink Executable Command:

plink.exe <users>@<IP or domain> -pw <password>
```

Figure 2 provides an example of a successful RDP tunnel created using Plink, and Figure 3 provides an example of communications being sent through the tunnel using port forwarding from the attacker C2 server.

Figure 2: Example of successful RDP tunnel created using Plink

Figure 3: Example of successful port forwarding from the attacker C2 server to the victim

It should be noted that for an attacker to be able to RDP to a system, they must already have access to the system through other means of compromise in order to create or

result of a payload dropped from a phishing email aimed at establishing a foothold into the environment, while simultaneously extracting credentials to escalate privileges. RDP tunneling into a compromised environment is one of many access methods typically used by attackers to maintain their presence in an environment.

Jump Box Pivoting

Not only is RDP the perfect tool for accessing compromised systems externally, RDP sessions can be daisy chained across multiple systems as a way to move laterally through an environment. FireEye has observed threat actors using the native Windows Network Shell (netsh) command to utilize RDP port forwarding as a way to access newly discovered segmented networks reachable only through an administrative jump box.

```
Example netsh Port Forwarding Command:
netsh interface portproxy add v4tov4 listenport
Example Shortened netsh Port Forwarding Command
netsh I p a v l=8001 listena=<JUMP BOX IP> conn
```

For example, a threat actor could configure the jump box to listen on an arbitrary port for traffic being sent from a previously compromised system. The traffic would then be forwarded directly through the jump box to any system

port forwarding gives threat actors a way to utilize a jump box's allowed network routes without disrupting legitimate administrators who are using the jump box during an ongoing RDP session. Figure 4 provides an example of RDP lateral movement to a segmented network via an administrative jump box.

Figure 4: Lateral Movement via RDP using a jump box to a segmented network

Prevention and Detection of RDP Tunneling

If RDP is enabled, threat actors have a way to move laterally and maintain presence in the environment through tunneling or port forwarding. To mitigate vulnerability to and detect these types of RDP attacks, organizations should focus on both host-based and network-based prevention and detection mechanisms. For additional information see the FireEye blog post on establishing a baseline for remote desktop protocol.

Host-Based Prevention:

- Remote Desktop Service: Disable the remote desktop service on all end-user workstations and systems for which the service is not required for remote connectivity.

- Local Accounts: Prevent the use of RDP using local accounts on workstations by enabling the "Deny log on through Remote Desktop Services" security setting.

Host-Based Detection:

Registry Keys:

- Review registry keys associated with Plink connections that can be abused by RDP session tunneling to identify unique source and destination systems. By default, both PuTTY and Plink store session information and previously connected ssh servers in the following registry keys on Windows systems:
  - HKEY_CURRENT_USER\Software\SimonTatham\PuTTY
  - HKEY_CURRENT_USER\SoftWare\SimonTatham\PuTTY\SshHostKeys
- Similarly, the creation of a PortProxy configuration with netsh is stored with the following Windows registry key:
  - HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\PortProxy\v4tov4
- Collecting and reviewing these registry keys can identify both legitimate SSH and unexpected tunneling

Event Logs:

- Review event logs for high-fidelity logon events. Common RDP logon events are contained in the following event logs on Windows systems:
  - %systemroot%\Windows\System32\winevt\Logs\Microsoft-TerminalServices-LocalSessionmanager%3Operational.evtx
  - %systemroot%\Windows\System32\winevt\Logs\Security.evtx
- The “TerminalServices-LocalSessionManager” log contains successful interactive local or remote logon events as identified by EID 21 and successful reconnection of a previously established RDP session not terminated by a proper user logout as identified by EID 25. The “Security” log contains successful Type 10 remote interactive logons (RDP) as identified by EID 4624. A source IP address recorded as a localhost IP address (127.0.0.1 – 127.255.255.255) may be indicative of a tunneled logon routed from a listening localhost port to the localhost’s RDP port TCP 3389.

Review your artifacts of execution for “plink.exe” file execution. Note that attackers can rename the file name to avoid detection. Relevant artifacts include, but are not limited to:

- Application Compatibility Cache/Shimcache

- Prefetch
- Service Events
- CCM Recently Used Apps from the WMI repository
- Registry keys

Network-Based Prevention:

- Remote Connectivity: Where RDP is required for connectivity, enforce the connection to be initiated from a designated jump box or centralized management server.
- Domain Accounts: Employ the “Deny log on through Remote Desktop Services” security setting for privileged accounts (e.g. domain administrators) and service accounts, as these types of accounts are commonly used by threat actors to laterally move to sensitive systems in an environment.

Network-Based Detection:

- Firewall Rules: Review existing firewall rules to identify areas of vulnerability to port forwarding. In addition to the potential use of port forwarding, monitoring for internal communications between workstations in the environment should be conducted. Generally, workstations do not have a need to communicate with one another directly and Firewall rules can be used to prevent any such communication, except where needed.

given port is what it appears to be. For example, threat actors may use TCP ports 80 or 443 to establish an RDP tunnel with a remote server. Deep inspection of the network traffic can likely reveal that it is not actually HTTP or HTTPS, but entirely different traffic all together. Therefore, organizations should closely monitor their network traffic.

- Snort Rules: The main indicator of tunneled RDP occurs when the RDP handshake has a designated low source port generally used for another protocol. Figure 5 provides two sample Snort rules that can help security teams identify RDP tunneling in their network traffic by identifying designated low source ports generally used for other protocols.

```
alert tcp any [21,22,23,25,53,80,443,8080] -> a
```

```
alert tcp any [21,22,23,25,53,80,443,8080] -> a
```

Figure 5: Sample Snort Rules to identify RDP tunneling

Conclusion

RDP enables IT environments to offer freedom and interoperability to users. But with more and more threat actors using RDP to move laterally across networks with limited segmentation, security teams are being challenged to decipher between legitimate and malicious RDP traffic. Therefore, adequate host-based and

malicious RDP usage.

Posted in Threat Intelligence

Related articles

Threat Intelligence

Hybrid Russian Espionage and Influence Campaign Aims to Compromise Ukrainian Military Recruits and Deliver Anti-Mobilization Narratives

By Google Threat Intelligence Group • 10-minute read

Threat Intelligence

Investigating FortiManager Zero-Day Exploitation (CVE-2024-47575)

By Mandiant • 19-minute read

How Low Can You Go? An Analysis of 2023 Time-to-Exploit Trends

By Mandiant • 10-minute read

capa Explorer Web: A Web-Based Tool for Program Capability Analysis

By Mandiant • 6-minute read