



## Decrypting RDP connections

The purpose of this lab is to give a taste of the power Wireshark has. In this lab, we will be working with RDP traffic. If one has the required key utilized between the two hosts for encrypting the traffic, Wireshark can deobfuscate the traffic for us.

When performing IR and analysis on Bob's machine, the IR team captured some PCAP of the RDP traffic they noticed from Bob's host to another host in the network. We have been asked to investigate the occurrence by our team lead. While combing his host for further evidence, you found an RDP-key hidden in a folder hive on Bob's host. After some research, we realize that we can utilize that key to decrypt the RDP traffic to inspect it.

Attempt to utilize the concepts from the Analysis Process sections to complete an analysis of the RDP-analysis.zip provided.

### Tasks:

#### Task #1

[Open the rdp.pcapng file in Wireshark.](#)

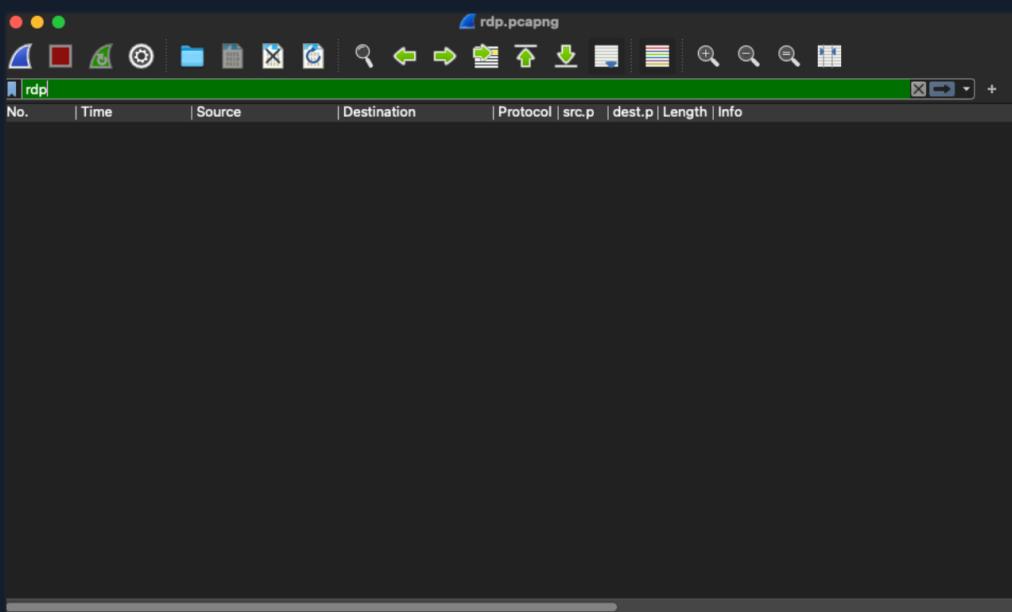
Unzip the zip file included in the optional resources and open it in Wireshark.

#### Task #2

[Analyze the traffic included.](#)

Take a minute to look at the traffic. Notice there is a lot of information here. We know our focus is on RDP, so let's take a second to filter on [rdp](#) and see what it returns.

### RDP Filter



Cheat Sheet

Resources

Go to Questions

### Table of Contents

#### Introduction

- Network Traffic Analysis
- Networking Primer - Layers 1-4
- Networking Primer - Layers 5-7

#### Analysis

- The Analysis Process
- Analysis in Practice

#### Tcpdump

- Tcpdump Fundamentals
- Capturing With Tcpdump (Fundamentals Labs)
- Tcpdump Packet Filtering
- Interrogating Network Traffic With Capture and Display Filters

#### Wireshark

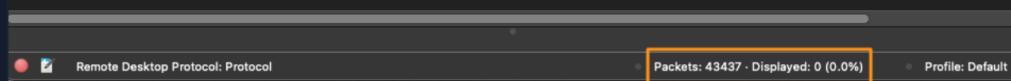
- Analysis with Wireshark
- Familiarity With Wireshark
- Wireshark Advanced Usage
- Packet Inception, Dissecting Network Traffic With Wireshark
- Guided Lab: Traffic Analysis Workflow
- Decrypting RDP connections

### My Workstation

O F F L I N E

Start Instance

∞ / 1 spawns left



As it stands, not much can be seen, right? This is because RDP, by default, is utilizing TLS to encrypt the data, so we will not be able to see anything that happened with RDP traffic. How can we verify its existence in this file? One way is to filter on the well-known port RDP uses typically.

Filter on port 3389 to determine if any RDP traffic encrypted or otherwise exists.

► Click to show answer

We can at least verify that a session was established between the two hosts over TCP port 3389.

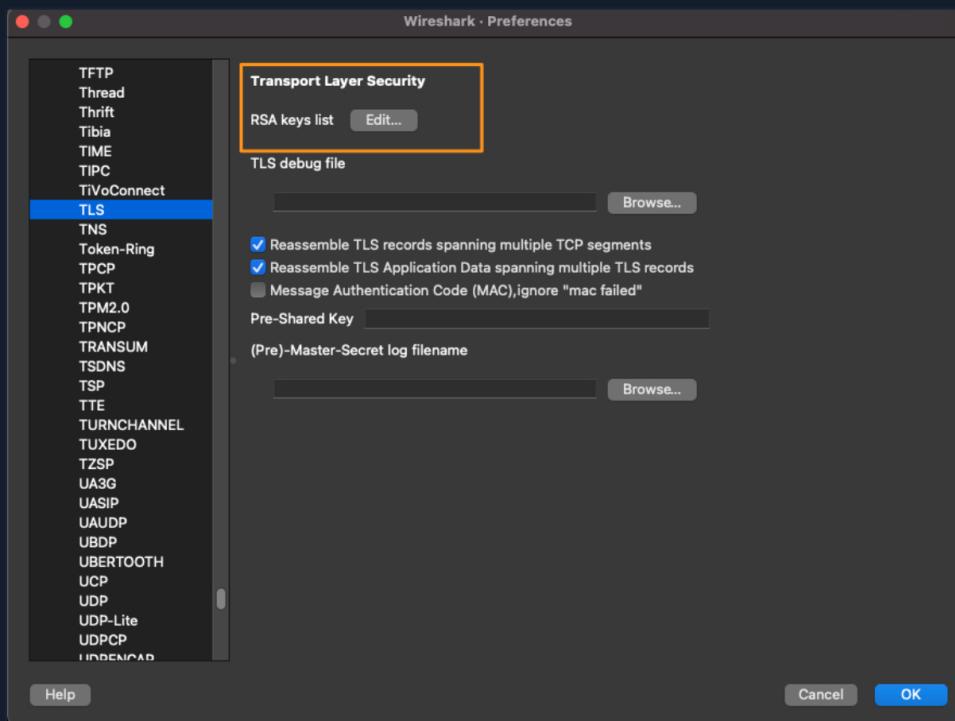
### Task #3

Provide the RDP-key to Wireshark so it can decrypt the traffic.

Now, let's take this a step further and use the key we found to try and decrypt the traffic.

To apply the key in Wireshark:

1. go to Edit → Preferences → Protocols → TLS
2. On the TLS page, select Edit by RSA keys list → a new window will open.



3. Follow the steps below to import the RSA server key.

### Import An RDP Key

#### Steps

Click the + to add a new key

Type in the IP address of the RDP server **10.129.43.29**

Type in the port used **3389**

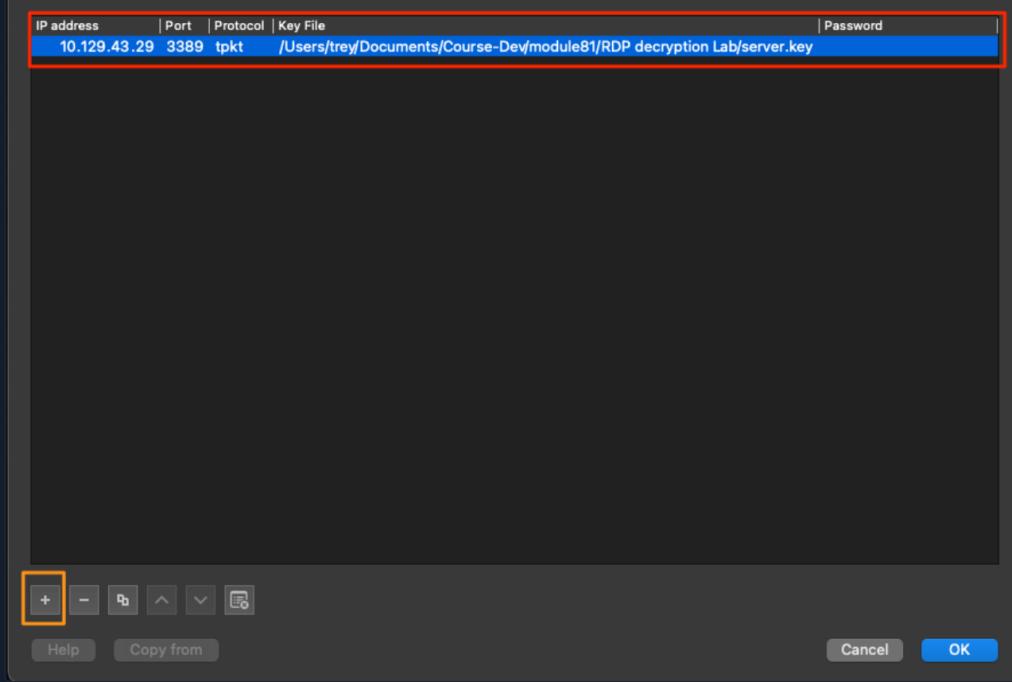
Protocol field equals **tpkt** or **blank**.

Browse to the **server.key** file and add it in the key file section.

Save and refresh your pcap file.

### Import Steps

TLS Decrypt



When filtering once again on RDP, we should see some traffic in the display.

### RDP In The Clear

No.	Time	Source	Destination	Protocol	src.p	dest.p	Length	Info
41	10.667508	10.129.43.27	10.129.43.29	RDP	506..	3389	545	ClientData
42	10.668282	10.129.43.29	10.129.43.27	RDP	3389	506..	289	ServerData Encryption: None (None)
61	10.675035	10.129.43.27	10.129.43.29	RDP	506..	3389	698	ClientInfo
62	10.678666	10.129.43.29	10.129.43.27	RDP	3389	506..	117	Error Alert
74	10.736304	10.129.43.29	10.129.43.27	RDP	3389	506..	555	Demand Active PDU
75	10.740888	10.129.43.27	10.129.43.29	RDP	506..	3389	762	Confirm Active PDU
76	10.741832	10.129.43.27	10.129.43.29	RDP	506..	3389	119	RDP PDU Type: Synchronize
78	10.741101	10.129.43.29	10.129.43.27	RDP	3389	506..	119	RDP PDU Type: Synchronize
79	10.741108	10.129.43.27	10.129.43.29	RDP	506..	3389	123	RDP PDU Type: Control, Action: Cooperate
80	10.741158	10.129.43.29	10.129.43.27	RDP	3389	506..	123	RDP PDU Type: Control, Action: Cooperate
81	10.741188	10.129.43.27	10.129.43.29	RDP	506..	3389	123	RDP PDU Type: Control, Action: Request control
83	10.741369	10.129.43.29	10.129.43.27	RDP	3389	506..	123	RDP PDU Type: Control, Action: Granted control
85	10.742142	10.129.43.27	10.129.43.29	RDP	506..	3389	133	RDP PDU Type: BitmapCache Persistent List
87	10.742188	10.129.43.27	10.129.43.29	RDP	506..	3389	123	RDP PDU Type: FontList
88	10.742284	10.129.43.29	10.129.43.27	RDP	3389	506..	123	RDP PDU Type: FontMap
89	10.742440	10.129.43.29	10.129.43.27	RDP	3389	506..	117	Virtual Channel PDU
91	10.742793	10.129.43.29	10.129.43.27	RDP	3389	506..	142	Virtual Channel PDU
93	10.742937	10.129.43.29	10.129.43.27	RDP	3389	506..	141	Virtual Channel PDU
94	10.742947	10.129.43.27	10.129.43.29	RDP	506..	3389	189	Virtual Channel PDU
96	10.743412	10.129.43.27	10.129.43.29	RDP	506..	3389	111	Virtual Channel PDU
98	10.744009	10.129.43.27	10.129.43.29	RDP	506..	3389	111	Virtual Channel PDU
99	10.747781	10.129.43.27	10.129.43.29	RDP	506..	3389	154	Virtual Channel PDU
101	10.748677	10.129.43.29	10.129.43.27	RDP	3389	506..	123	Virtual Channel PDU
102	10.748738	10.129.43.29	10.129.43.27	RDP	3389	506..	155	Virtual Channel PDU
103	10.748781	10.129.43.29	10.129.43.27	RDP	3389	506..	152	Virtual Channel PDU
105	10.748838	10.129.43.29	10.129.43.27	RDP	3389	506..	149	Virtual Channel PDU
107	10.751521	10.129.43.29	10.129.43.27	RDP	3389	506..	148	Virtual Channel PDU

From here, we can perform an analysis of the RDP traffic. We can now follow TCP streams, export any potential objects found, and anything else we feel necessary for our investigation. This works because we acquired the RSA key used for encrypting the RDP session. The steps for acquiring the key were a bit lengthy, but the short of it is that if the RDP certificate is acquired from the server, [OpenSSL](#) can pull the private key out of it.

## Perform Analysis of the Unencrypted Traffic

Now that we have broken RDP out of the TLS tunnel, what can we find? Perform the analysis steps and attempt to answer the questions below.

### Questions:

What host initiated the RDP session with our server?

► [Click to show answer](#)

Which user account was used to initiate the RDP connection?

► [Click to show answer](#)

## Summary:

This lab was to serve as an example of what Wireshark can do with captured data and its plugins. Wireshark's capability to ingest information and illuminate the obscure is robust. Having the ability to decrypt data after ingestion is a powerful capability. This concept could be applied to any protocol that utilizes encryption as long as we have the key that will be utilized to establish the connections.

### VPN Servers

**⚠ Warning:** Each time you "Switch", your connection keys are regenerated and you must re-download your VPN connection file.

All VM instances associated with the old VPN Server will be terminated when switching to a new VPN server.

Existing PwnBox instances will automatically switch to the new VPN server.

US Academy 3

Medium Load ▾

### PROTOCOL

UDP 1337    TCP 443

[DOWNLOAD VPN CONNECTION FILE](#)



### Connect to Pwnbox

Your own web-based Parrot Linux instance to play our labs.

### Pwnbox Location

UK

139ms ▾

**!** Terminate Pwnbox to switch location

[Start Instance](#)

∞ / 1 spawns left

Waiting to start...



Enable step-by-step solutions for all questions **?** ⚡

## Questions

Answer the question(s) below to complete this Section and earn cubes!



Cheat Sheet

[Download VPN](#)

Target(s): [Click here to spawn the target system!](#)

Connection File

+ 2 📁 What user account was used to initiate the RDP connection?

bucky

 Submit

 Hint

 Previous

 Finish

Powered by 