

## Frågor

Besvara följande frågor genom att studera datan med hjälp av Spark och Databricks.

### 1. Vilka protokoll finns representerade i datan?

TCP? UDP? ICMP? RSVP? QUIC?

```
protocols = data.select("protocol_type").distinct()
protocols.show()
```

▶ protocols: pyspark.sql.dataframe.DataFrame = [protocol\_type: string]

```
+-----+
|protocol_type|
+-----+
|      tcp    |
|      udp    |
|      icmp   |
+-----+
```

### 2. Hur många tjänster (services) är det som använder protokoll...

TCP? UDP? ICMP? RSVP? QUIC?

```
service_count = data.groupBy("protocol_type", "service").count()
service_count.show()
```

▶ service\_count: pyspark.sql.dataframe.DataFrame = [protocol\_type: string, service: string ... 1 more field]

```
|      tcp|  csnet_ns|  126|
|      tcp|   finger|  670|
|      udp| domain_u| 5863|
|      tcp|    IRC   |   43|
|      tcp|    time  |  157|
|      tcp|    link  |  102|
|      tcp|   kshell |   98|
|      tcp| uucp_path|  106|
|      tcp| netbios_ns| 102|
|      tcp|   gopher |  117|
|      tcp|   whois  |  110|
|      tcp|    ctf   |   97|
|      tcp| netstat  |   95|
|      tcp|    name  |   98|
|      tcp|    auth  |  328|
|      tcp| remote_job|  120|
|      tcp|  sql_net |  110|
|      tcp| netbios_dgm|  99|
+-----+-----+-----+
```

only showing top 20 rows


### 3. Sökning i paketklassifikation

Paketen i datan har genomgått ett analysverktyg som har klassat dem som normala, eller som en del av en attack. Denna information går att hitta under kolumnen "label". Paket som är klassade som attack-data har en label som inte är "normal."

#### 3.1. Hur många paket är klassade som en del av en attack, och använder protokoll...

TCP? UDP? ICMP? RSVP? QUIC?

```
attack_packets = data.filter(data["label"] != "normal.")
attack_packets.groupBy("protocol_type").count().show()
```

▶  attack\_packets: pyspark.sql.dataframe.DataFrame = [duration: integer, protocol\_type: string ... 40 more fields]

```
+-----+-----+
|protocol_type| count|
+-----+-----+
|          tcp|113252|
|          udp|  1177|
|          icmp|282314|
+-----+-----+
```

#### 3.2. Hur många procent av det totala antalet paket (avrundat till hela procent) har en label som inte är "normal."?

```
Total data packets:  494021
Total attacked data packets:  396743
Percentage of attack packets: 80%
```

```
total_packets = data.count()
print("Total data packets: ", total_packets)

total_attack_packets = attack_packets.count()
print("Total attacked data packets: ", total_attack_packets)

attack_percentage = (total_attack_packets / total_packets) * 100
print(f"Percentage of attack packets: {round(attack_percentage)}%")
```

### 3.3. Hur många ICMP-paket är klassade som en del av en attack, men är inte en så kallad "smurf"-attack?

```
icmp_attack_packets = attack_packets.filter(attack_packets["protocol_type"] == "icmp")

# Exclude 'smurf' attacks
non_smurf_icmp_attacks = icmp_attack_packets.filter(icmp_attack_packets["label"] != "smurf")
non_smurf_icmp_attacks_count = non_smurf_icmp_attacks.count()
print(f"Non-smurf ICMP attack packets: {non_smurf_icmp_attacks_count}")
```

```
▶ icmp_attack_packets: pyspark.sql.dataframe.DataFrame = [duration: integer, protocol_type: string ... 40 more fields]
▶ non_smurf_icmp_attacks: pyspark.sql.dataframe.DataFrame = [duration: integer, protocol_type: string ... 40 more fields]
```

Non-smurf ICMP attack packets: 282314