Tunnelling

Tunnelling Traffic : ICMP and DNS

Traffic tunnelling is (also known as "port forwarding" transferring the data/resources in a secure method to network segments and zones.

It can be used for "internet to private network" and "private network to internet" flow/direction.

There is an encapsulation process to hide the data, so the transfered data appear ~~naturally~~ natural for the case, but it contain private data packets and transfers them to the final destination securely.

So Basically network tunnelling is hiding one kind of traffic inside another kind of traffic so it can cross network and firewalls as if it were "normal" allowed traffic

Tunnelling = encapsulation

We take "real" data packets and wrap them inside another protocol's packets

Like putting a letter inside another envelope (ICMP/DNS/HTTPS), so outsiders mostly see the outer envelope

So it helps security and confidentiality and Connectivity

- the inside data is often encrypted
- let us safely reach private network from the internet or link private networks togethes

How tunnelling is Been used By attackes:

- If the firewall allows protocols like DNS and ICMP, attackers can smuggle C2 through those protocols using DNS tunnelling or ICMP tunnelling bypassing security rules that would Block

So we need to spot when DNS/ICMP traffic doesn't look normal

Difference

- Port forwarding : mapping a port from one place to anothes
  (if traffic comes here, send it there)
  (forwarding port 22 to internal SSH server)

- Tunnelling : encapsulating traffic inside anothes protocol

What ICMP do: (Internet Control Message Protocol)

- Reports problems when IP packets can't be delivered
- Helps diagnose connectivity
    - ping uses ICMP Request/Echo Reply
    - traceroute uses ICMP messages to show the path/hops

- It's not used to carry application data like HTTP
- It's mostly control/feedback messages about network

So attackers hide commands or stolen data inside the ICMP payload. If a firdwall allows ICMP, it may look like normal ping traffic and get through. But this only matters when there is already malware inside the network that can read those hidden commands and send data back out.

Sneaking out information

Detection:

- High volume over time
- weird external destination
- Very regulas timing
- Paylodd that looks like random data instead of simple pattesn
- Large ~~sz~~ ICMP Echo packets
- Big size

Wireshark

Global search - icmp

- Packet length - data.len > 64 and icmp
- ICMP destination address
- Encapsulated protocol sign in ICMP payload

DNS Analysis

DNS tunnelling is very similar with ICMP

ICMP : hidden bytes are in the ICMP/ data/payload

DNS : hidden bytes are in
- the subdomain part of the query
- TXT records in responses
- sometimes unusual record types or patters

Detection :
- Long subdomains
- High query rate to a single domain
- Lots of NXDomain responses (queries for names that doesn't exist)
- Heavy use of TXT queries and response
-

Wireshark

Global search dns

Query length

Anomalous and non-regular names in DNS addresses — dns contains "dnscat"

Long DNS with encoded subdomain addresses — dns.qry.name.len >15 and !mdns

Known pattern like dnscat and dns2tcp

!mdns : Disable local link device queries

Questions

1) Investigate the anomalous packet. Which protocol is used in ICMP tunnelling?

data.len >126 and icmp

Then spotted unusual 886 echo ICMP and found the answes from the encode

2) what is the suspisious main domain address that receives anomalous DNS queries?

dns.qry.name.len >30 and !mdns

Because it says receives