Ethical Student Hackers

Understanding and Exploring Networks

SHEFFIELD Ethical Student Hackers
Breaking into security.

The Legal Bit

- The skills taught in these sessions allow identification and exploitation of security vulnerabilities in systems. We strive to give you a place to practice legally, and can point you to other places to practice. These skills should not be used on systems where you do not have explicit permission from the owner of the system. It is VERY easy to end up in breach of relevant laws, and we can accept no responsibility for anything you do with the skills learnt here.

- If we have reason to believe that you are utilising these skills against systems where you are not authorised you will be banned from our events, and if necessary the relevant authorities will be alerted.

- Remember, if you have any doubts as to if something is legal or authorised, just don't do it until you are able to confirm you are allowed to.

SHEFFIELD Ethical Student Hackers
Breaking into security.

Code of Conduct

- Before proceeding past this point you must read and agree to our Code of Conduct - this is a requirement from the University for us to operate as a society.
- If you have any doubts or need anything clarified, please ask a member of the committee.
- Breaching the Code of Conduct = immediate ejection and further consequences.
- Code of Conduct can be found at https://shefesh.com/downloads/SESH%20Code%20of%20Conduct.pdf

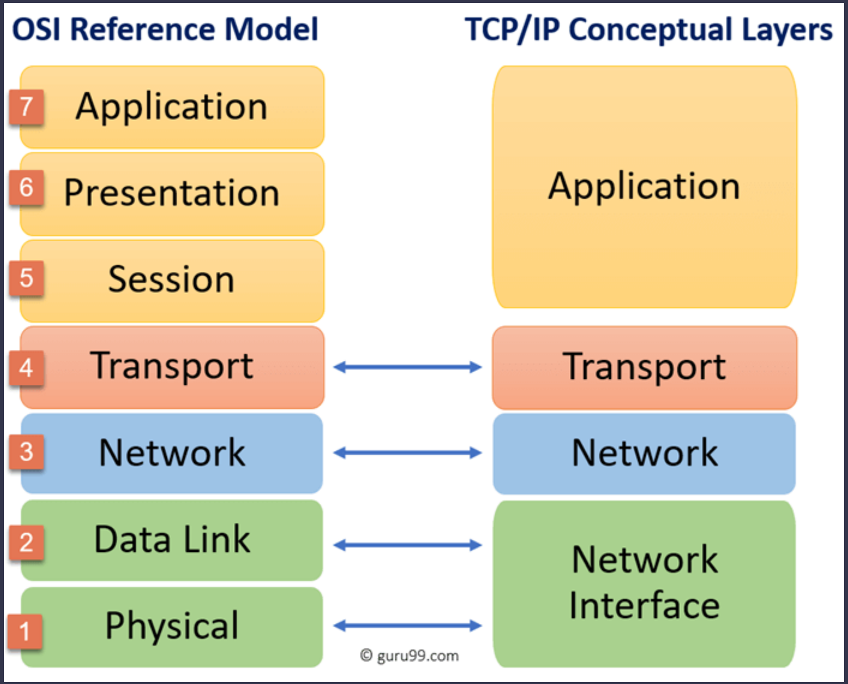
SHEFFIELD Ethical Student Hackers
Breaking into security.

Why Learn Networking?

- Computer networks are ubiquitous; every day, more devices come online
- While this opens up countless possibilities, it also means that networked devices need to defend against remote threats – not just local ones
- At their core, security holes are design oversights. To find what others haven't thought of, you'll need to understand everything they have
- By learning networking, you'll learn where there are holes and how to patch / exploit them

The (Simplified) OSI Model

1. Physical (wires, radio, optical)
2. Data Link (Ethernet, 802.11, ARP, etc)
3. Network (IPv4/IPv6, ICMP, WireGuard, etc)
4. Transport (TCP, UDP)
5. Application (HTTP, SSH, DHCP, SMB, etc)



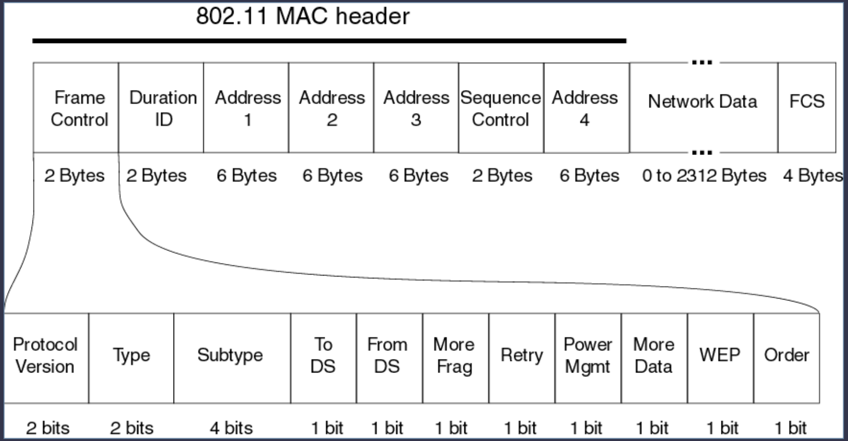
The Physical Layer

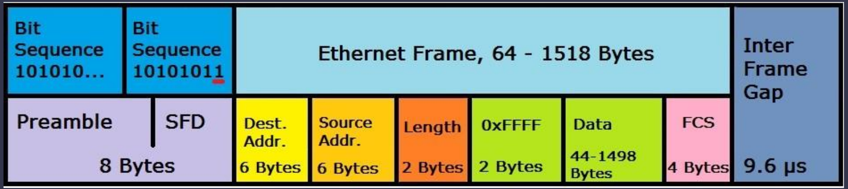
- This is the simplest layer of a network and is only concerned with the transmission of raw information
- Cables often have “link lights” that indicate if they are functional
- Wireless access points often have something similar
- As most network attacks are carried out remotely, this layer isn’t the most interesting to us
- With that being said, some connections (particularly wireless ones) can be physically jammed

Sheffield Ethical Student Hackers
Breaking into security.

The Data Link Layer

- The dominant protocols at this layer are Ethernet and WiFi (802.11)
- At this level, MAC addresses are used to identify different machines on the same network
- Ethernet is generally unsecured, so access to Ethernet ports needs to be tightly controlled
- Access can be obscured via MAC filtering
- VLANs can be used to isolate parts of the network
- There are many ways to secure WiFi: WEP, WPS, WPA2-PSK, WPA2-Enterprise, etc.





WiFi Deauth Attacks

- The 802.11 standard for WiFi allows for cleartext “management” frames. One of these frames is designed to “deauthenticate” a station and kick them off of the network
- This can be used to soft-jam a WiFi network, capture a WPA handshake, or generate ARP requests
- This can be performed using any wireless NIC capable of monitor mode and packet injection

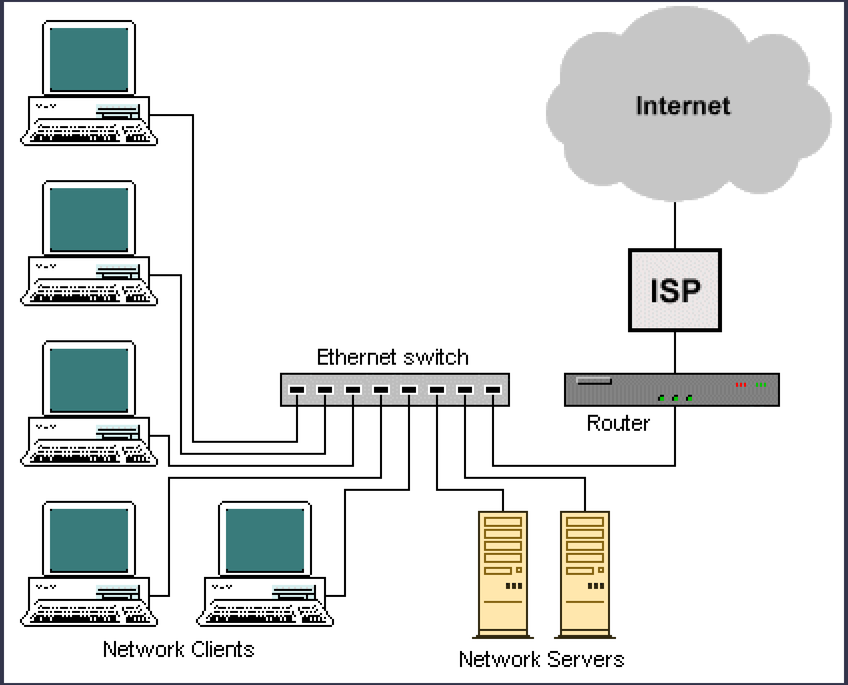
```
airmon-ng start wlan0

aireplay-ng -0 1 -a 00:14:6C:7E:40:80 -c 00:0F:B5:34:30:30 wlan0
```

- `-0` is the flag for deauthentication (`1` is the number of deauths)
- `-a 00:14:6C:7E:40:80` is the MAC address of the AP we are impersonating
- `-c 00:0F:B5:34:30:30` is the MAC of the client to deauth (can be left blank for broadcast)
- `wlan0` is the name of the network interface to use

SHEFFIELD Ethical Student Hackers
Breaking into security.

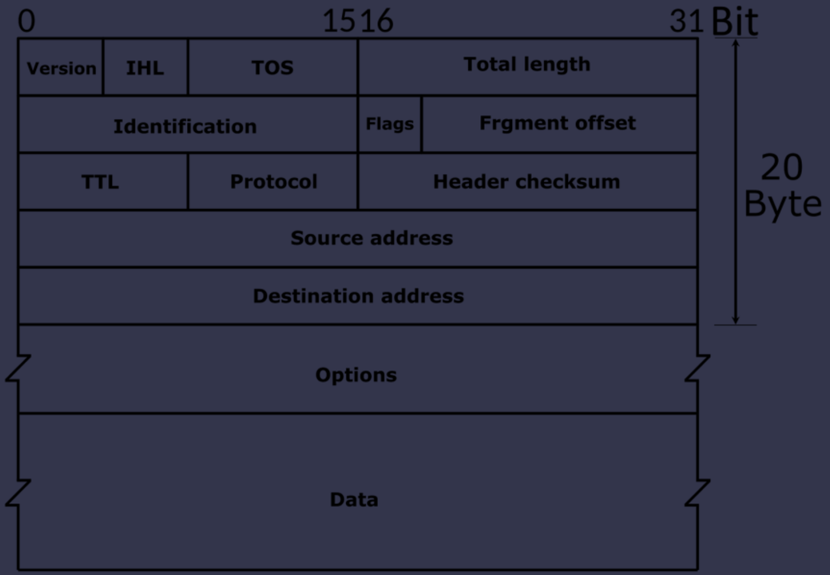
The Network Layer

- While MAC addresses are fine to messaging within a LAN, the limited addresses and broadcast nature of Ethernet means a different protocol is needed to connect to the internet
- There are two versions of IP: IPv4 and IPv6
- IPv4 is still the most commonly used, but is being steadily replaced by IPv6
- IP is designed to perform a best-effort routing of packets from A to B, but does not guarantee delivery, order, or duplicate avoidance



IPv4 Packet Structure

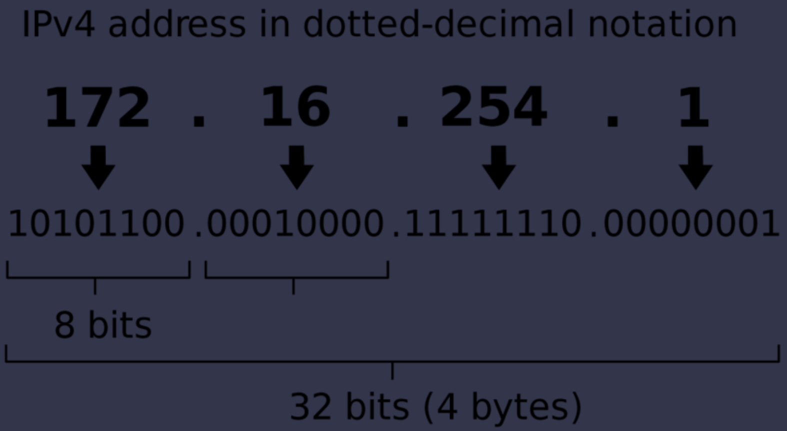
Version – Always 4 for IPv4
IHL – The length of the header (the "options" field is optional)
TOS – Type of service; used in VoIP to prioritise packets
Total Length – Combined size of the header and data
Identification – Identify each fragment of a packet
Flags – Are fragments allowed? More to come?
Fragment Offset – Where does this fragment start?
TTL – How many hops before this packet is tossed out?
Protocol – ICMP (1), TCP (6), UDP (17)
Header Checksum – For checking the validity of the header
Src / Dest Addresses – Where from and where to?
Options – Optional extra information
Data – The actual information to transmit



Breaking into security.

IPv4 Addresses

- Every computer on the Internet has a unique IPv4 address* that it can use to send and receive data
- These are broken into subnets of different sizes. Computers can communicate directly within them, but a router is needed to bridge between them
- CIDR notation is used to communicate the size of a subnet: 192.168.0.0/16
- Subnet address
- Subnet size (in bits)
- All addresses in a subnet contain the subnet address with the remaining bits free to change
- There are some common, reserved subnets:
    - 0.0.0.0/8 – the current network (source only)
    - 127.0.0.0/8 – the loopback / localhost address
    - 10.0.0.0/8 – large private networks
    - 192.168.0.0/16 – smaller private networks



* Except when they don't

Sheffield Ethical Student Hackers
Breaking into security.

Routing & ARP

- How does IP help us route data from one computer to the next?
  - Every computer maintains its own routing table that maps subnets to interfaces and, when the address isn't local, a forwarding gateway
  - Routes are tried in order **from most specific to least specific**
- When the address is **NOT** on the LAN:
  - The packet is forwarded to the default gateway which has an interface for forwarding to the public internet
- When the address **is** on the LAN:
  - The routing table indicates which interface the subnet is present on
  - From there, ARP (IPv4) or ND (IPv6) maps IP addresses to MAC addresses
  - Ethernet or 802.11 frames can then be used to transmit the packet to its correct destination
- ARP Poisoning (or similar route spoofing) allow for Man in the Middle (MitM) attacks, where traffic is routed through an attacker's computer

Poking Around

ip addr – Get a list of all available interfaces and their address information

ip neigh – Get the currently cached ARP table (find the MAC addresses of local IPs)

ip route – Get the current routing table, showing where each subnet is connected

nmap -sn 192.168.42.0/24 – Scan all addresses of a given subnet, but skip the port scan (ping scan)

ping 10.0.0.6 – Check if 10.0.0.6 is reachable* and how long a round-trip takes

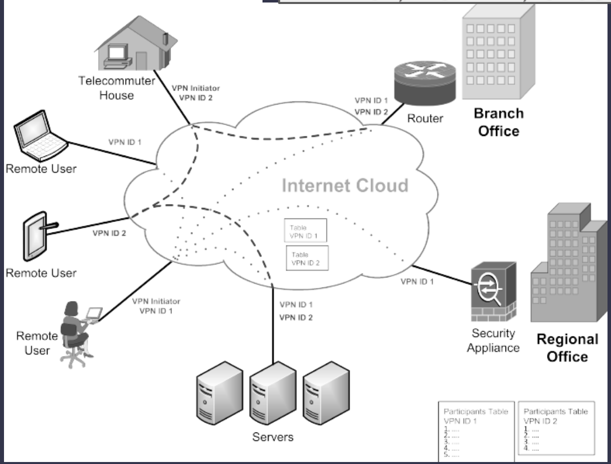
traceroute 192.168.4.1 – Path taken to get from the local machine to the destination address

* Usually, in the absence of any tomfoolery

SHEFFIELD Ethical Student Hackers
Breaking into security.

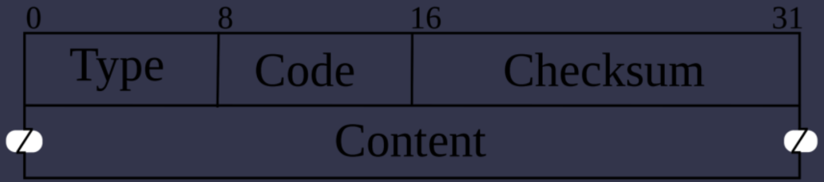
VPNs Are Remote LANs

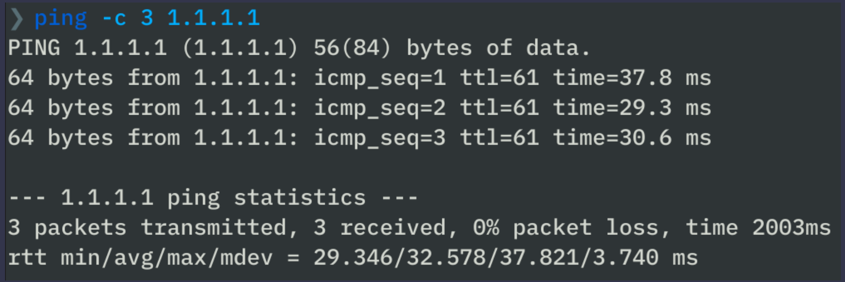
- VPNs work at the network level to build fake LANs between remote networks
- They allow access to private services within the network that can't be accessed from the Internet directly
- Additionally, most VPNs encrypt all of the traffic sent through them. This can improve privacy and security, but only if the endpoint is to be trusted
- While all LAN-traffic used to be publicly visible, nowadays nearly all important data (e.g. banking details) are already encrypted via HTTPS **without** needing to use a VPN



Internet Control Message Protocol

- ICMP is used to relay important control and status messages over IP
- Type 8 code 0 (8:0) signals a ping request, to which clients can respond with 0:0 (a ping response)
- When a packet's TTL expires, an ICMP 11:0 message is used to inform the sender – this is how traceroute works
- ICMP can be used to probe a network or even launch a DoS attack. As a result, some admins take the blunt approach of disabling ICMP, but this is only **security through obscurity**





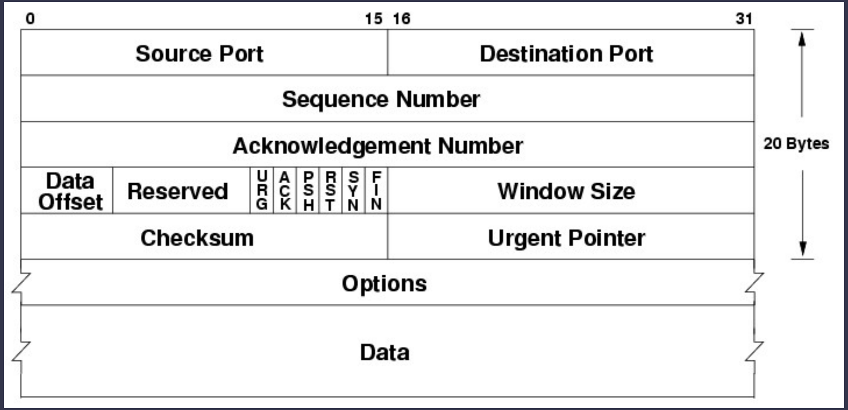
```
❯ ping -c 3 1.1.1.1
PING 1.1.1.1 (1.1.1.1) 56(84) bytes of data.
64 bytes from 1.1.1.1: icmp_seq=1 ttl=61 time=37.8 ms
64 bytes from 1.1.1.1: icmp_seq=2 ttl=61 time=29.3 ms
64 bytes from 1.1.1.1: icmp_seq=3 ttl=61 time=30.6 ms

--- 1.1.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 29.346/32.578/37.821/3.740 ms
```

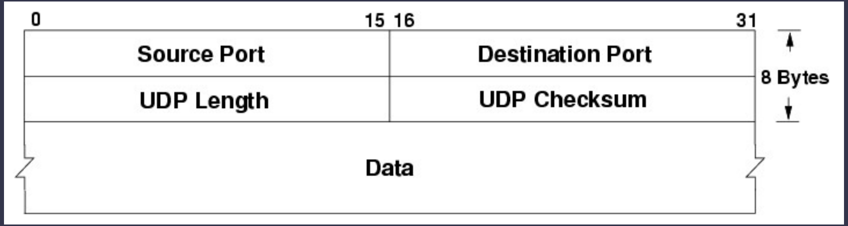
https://blog.securityevaluators.com/icmp-the-good-the-bad-and-the-ugly-130413e56030

SHEFFIELD Ethical Student Hackers
Breaking into security.

The Transport Layer

- While IP is great for routing packets, it’s still missing some features:
    - How do you know which application a packet is intended for?
    - How do you ensure reliable transmission?
- TCP provides reliable transmission by establishing a connection, requiring acknowledgement, and ordering packets
- UDP still allows for multiplexing and targeting via port numbers, but is used for quick, **unreliable** transport
- Services like HTTP use TCP while UDP is used for VPNs and things like VoIP

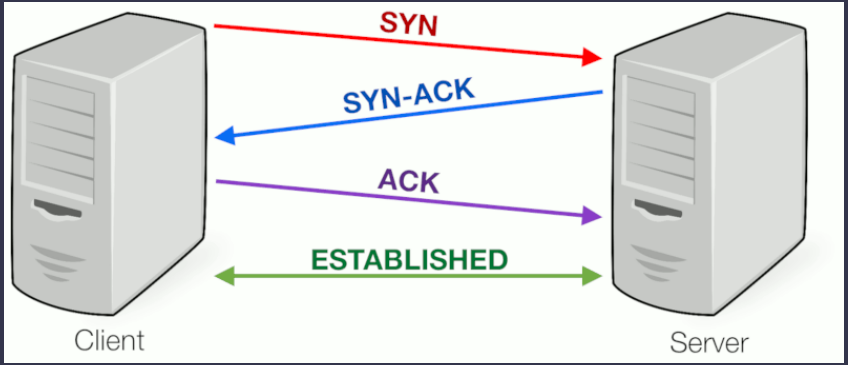


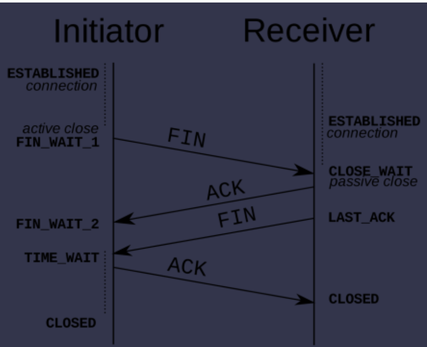


Sheffield Ethical Student Hackers
Breaking into security.

Transport Control Protocol (TCP)

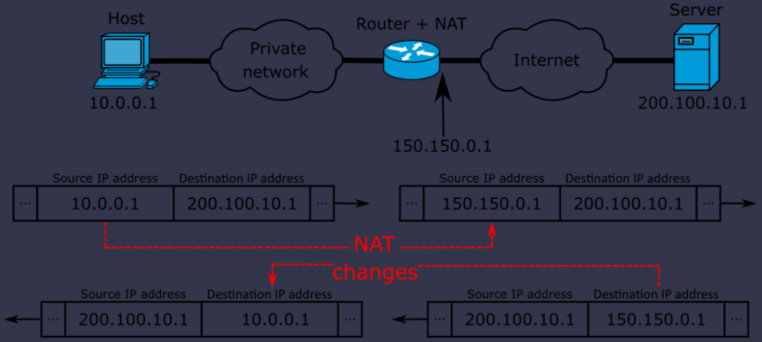
- TCP builds stateful connections and requires a handshake to both set up and tear down
- When packets are lost, they can be re-requested
- A SYN-flood is a form of DoS attack that creates thousands of hanging connections by sending SYN requests with spoofed source addresses

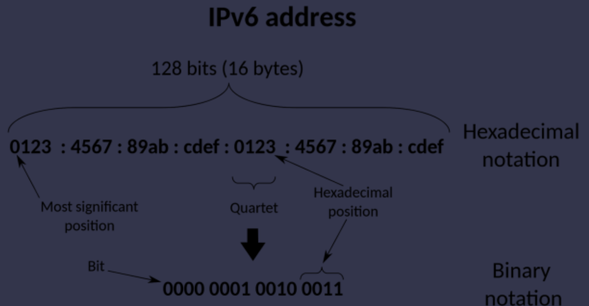




Network Address Translation (NAT)

- There are ~4 billion IPv4 addresses; that's a lot, but not enough for every device on the internet
- IPv6 solves this with longer, 128 bit addresses (allowing for 3.4×10^{38} unique IPs), but still hasn't been widely adopted
- NAT allows for local addresses to be non-unique (within the private subnets) but to be "routed" through a single, shared address
- A table of TCP / UDP ports is used to associate incoming connections with local devices
- NAT acts like a weak firewall, denying unrequested incoming packets by default, but is usually only an inconvenience for the attacker as there are many ways to "punch-through" it





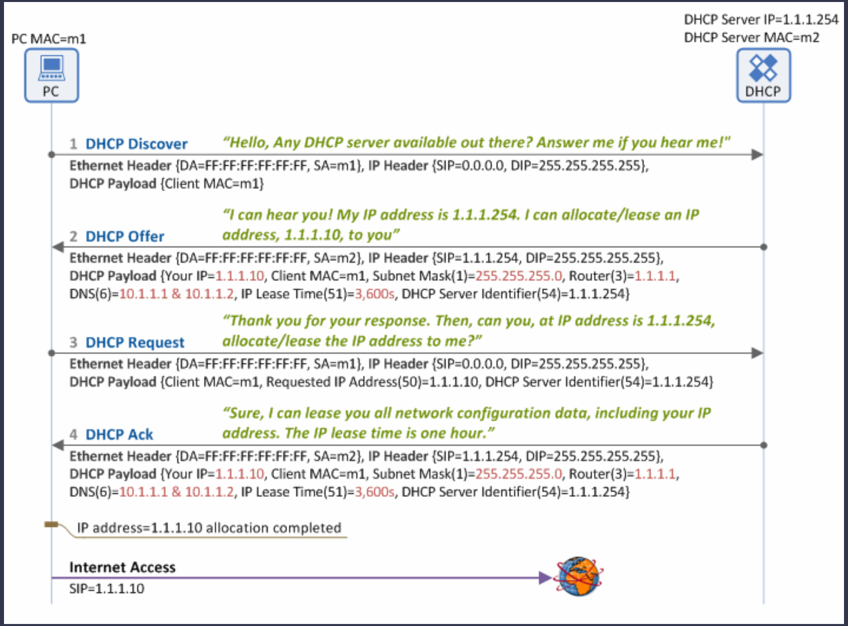
Higher Level Protocols

SHEFFIELD Ethical Student Hackers

Breaking into security.

Using DHCP to Let Local IPs

- While it's possible to manually set IP addresses for every client in a local network, this requires planning and is prone to collisions
- DHCP is a high-level protocol that runs atop UDP is is used to allocate IP addresses to devices on the network
- Rogue DHCP servers on a network can trick clients into using a malicious DNS service or create intentional address clashes



1 **DHCP Discover** *"Hello, Any DHCP server available out there? Answer me if you hear me!"*
Ethernet Header {DA=FF:FF:FF:FF:FF:FF, SA=m1}, IP Header {SIP=0.0.0.0, DIP=255.255.255.255},
DHCP Payload {Client MAC=m1}

2 **DHCP Offer** *"I can hear you! My IP address is 1.1.1.254. I can allocate/lease an IP address, 1.1.1.10, to you"*
Ethernet Header {DA=FF:FF:FF:FF:FF:FF, SA=m2}, IP Header {SIP=1.1.1.254, DIP=255.255.255.255},
DHCP Payload {Your IP=1.1.1.10, Client MAC=m1, Subnet Mask(1)=255.255.255.0, Router(3)=1.1.1.1,
DNS(6)=10.1.1.1 & 10.1.1.2, IP Lease Time(51)=3,600s, DHCP Server Identifier(54)=1.1.1.254}

3 **DHCP Request** *"Thank you for your response. Then, can you, at IP address is 1.1.1.254, allocate/lease the IP address to me?"*
Ethernet Header {DA=FF:FF:FF:FF:FF:FF, SA=m1}, IP Header {SIP=0.0.0.0, DIP=255.255.255.255},
DHCP Payload {Client MAC=m1, Requested IP Address(50)=1.1.1.10, DHCP Server Identifier(54)=1.1.1.254}

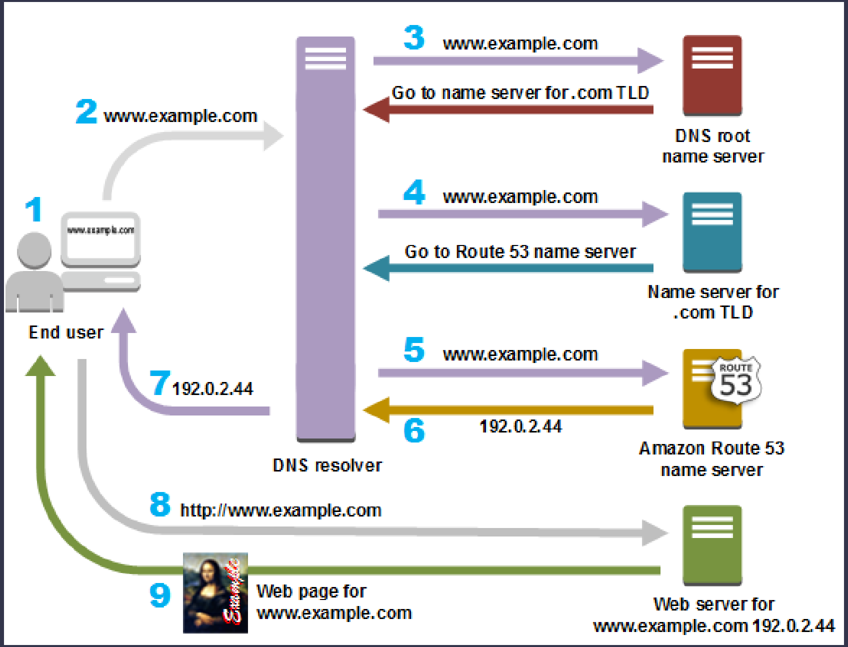
4 **DHCP Ack** *"Sure, I can lease you all network configuration data, including your IP address. The IP lease time is one hour."*
Ethernet Header {DA=FF:FF:FF:FF:FF:FF, SA=m2}, IP Header {SIP=1.1.1.254, DIP=255.255.255.255},
DHCP Payload {Your IP=1.1.1.10, Client MAC=m1, Subnet Mask(1)=255.255.255.0, Router(3)=1.1.1.1,
DNS(6)=10.1.1.1 & 10.1.1.2, IP Lease Time(51)=3,600s, DHCP Server Identifier(54)=1.1.1.254}

IP address=1.1.1.10 allocation completed

Internet Access
SIP=1.1.1.10

Resolving Names with DNS

- All of the protocols we've seen so far rely on having the IP address of the computer we are communicating with, so how does typing google.co.uk in your address bar get you anywhere?
- The DNS mapping is cached on both local devices and your DNS resolver, so only rarely will you need to go through the whole resolution process
- Like with ARP, these caches can be poisoned (by impersonating a name server) and victims could be unknowingly redirected to a malicious IP



Having Another Nosy

ss -tup – List all TCP and UDP connections, as well as their process

ss -tupl – List all TCP and UDP connections and processes listening for new connections

ss -s – Get a summary of the system's current connections

dig shefesh.com – Perform a DNS lookup of shefesh.com, returning its IP address

dig -x 8.8.8.8 – Perform a reverse DNS lookup of 8.8.8.8, returning its domain names

SHEFFIELD Ethical Student Hackers
Breaking into security.

Sniffing Some Packets

What is “packet sniffing”?

- Packet sniffing is the interception and analysis of network traffic
- Because of the broadcast nature of the link-level protocols (Ethernet and 802.11), it’s possible to see not just your own traffic, but **all** traffic on the local network

What is Wireshark?

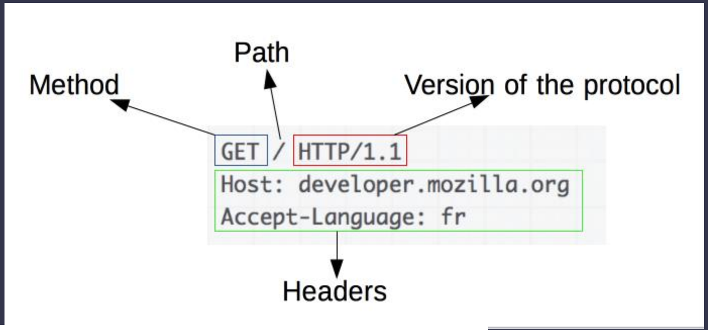
- Wireshark is a popular, cross-platform application for packet sniffing

What can it do?

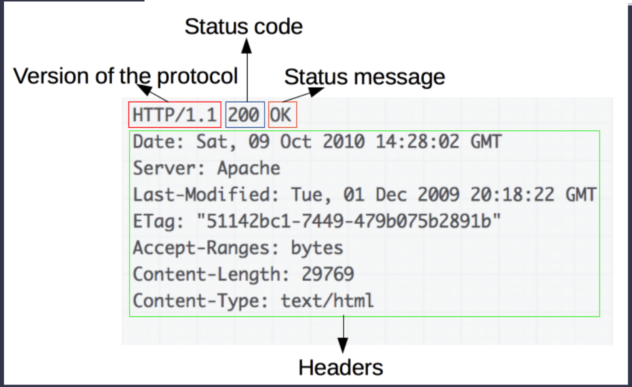
- Wireshark can provide information about every bit of data traversing the network, letting you inspect Ethernet frames, IP packets, TCP segments, and more
- For connection-based protocols, conversations can be reassembled and the transmitted files extracted

Understanding HTTP

- HTTP is a stateless protocol that allows for the communication between a client and server via a request-response model
- The request method is most commonly GET or POST, but many others like HEAD and PUT, also exist
- There are many response status codes, but here are a few of the more common ones:
  - 200 – OK (request successful)
  - 301 – Moved Permanently
  - 404 – Not Found (path didn't exist)
  - 500 – Internal Server Error
- After the HTTP headers, there is an optional body in which the content is actually stored



```
GET / HTTP/1.1
Host: developer.mozilla.org
Accept-Language: fr
```



```
HTTP/1.1 200 OK
Date: Sat, 09 Oct 2010 14:28:02 GMT
Server: Apache
Last-Modified: Tue, 01 Dec 2009 20:18:22 GMT
ETag: "51142bc1-7449-479b075b2891b"
Accept-Ranges: bytes
Content-Length: 29769
Content-Type: text/html
```

SHEFFIELD Ethical Student Hackers
Breaking into security.

Common HTTP Headers

General Headers:

- Date – The date and time at which the message originated
- Connection – Often set to Keep-Alive when the connection is to be reused
- Content-Length – Size of the body in bytes
- Content-Type – Specifies how the body should be interpreted

Request Headers:

- User-Agent – Describes the client making the request
- Accept – Specify the content types that will be accepted as a response
- Cookie – Used to communicate session information back to the server

Response Headers:

- Server – Information about the server that handled the request

SHEFFIELD Ethical Student Hackers
Breaking into security.

HTTP POST Requests

Sending URL Encoded Data

```
POST /test HTTP/1.1
Host: foo.example
Content-Type: application/x-www-form-urlencoded
Content-Length: 27

field1=value1&field2=value2
```

Sending Raw Multipart Data

```
POST /test HTTP/1.1
Host: foo.example
Content-Type: multipart/form-data;boundary="boundary"

--boundary
Content-Disposition: form-data; name="field1"

value1
--boundary
Content-Disposition: form-data; name="field2"; filename="example.txt"

value2
--boundary--
```

HTTP Cookies

- Cookies are a way to emulate state over the stateless HTTP protocol and can be used to for authentication or storing other session information
- Cookies are set by the server via an HTTP response header. The client then stores these cookies and sends them in future HTTP requests
- Authentication cookies are particularly valuable tidbits as a valid authentication cookie is often all that's needed to impersonate another user
- Private cookies can be extracted using methods like XSS

```
HTTP/2.0 200 OK
Content-Type: text/html
Set-Cookie: yummy_cookie=choco
Set-Cookie: tasty_cookie=strawberry

[page content]
```

```
GET /sample_page.html HTTP/2.0
Host: www.example.org
Cookie: yummy_cookie=choco; tasty_cookie=strawberry
```

SHEFFIELD Ethical Student Hackers
Breaking into security.

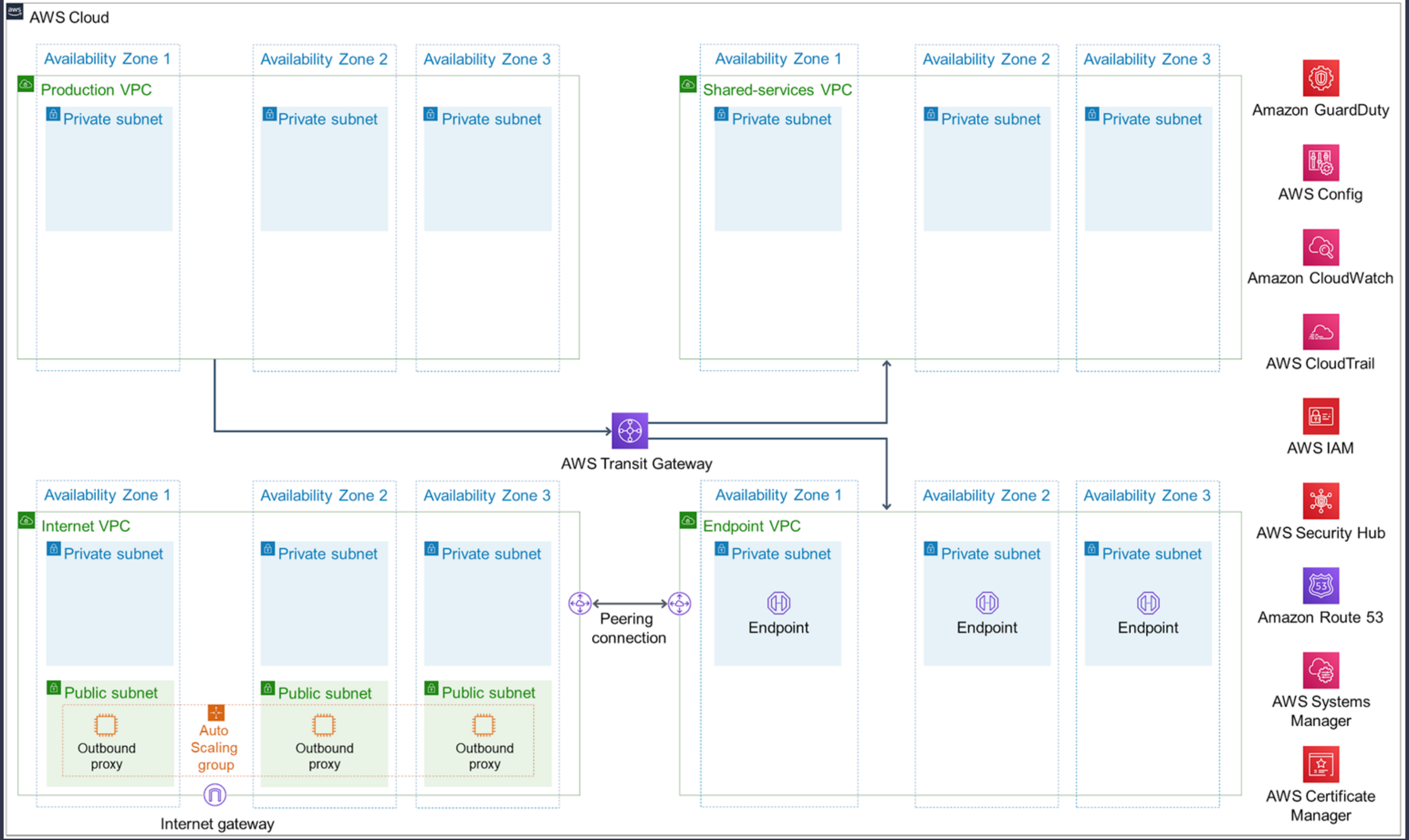
Secure Networking

The 'Security by Layers' principle

- Why have one barrier when you can have multiple?
- Protect your network with firewalls and multiple levels of authentication, based on the sensitivity of what is in that part of the network
- A 'layered approach' may also refer to employing multiple techniques on different services - firewalls, intrusion detection systems, email & web filtering, packet inspection, and encryption

Separate subnets

- Partition your network into publicly visible, internet-accessible IP ranges; and IP ranges that are only accessible by other machines on your network
- Create 'jumpboxes' or 'Bastion servers' that guard the way into the network
- These 'partitioned' networks are a key component of Secure Cloud Infrastructure - for example, AWS' Virtual Private Clouds (VPCs)



An example of a highly-rated network architecture
https://aws.amazon.com/quickstart/architecture/compliance-uk-official/

SHEFFIELD Ethical Student Hackers
Breaking into security.

Security Bits & Bobs

Air gapping

- A technique to physically separate a machine or subnet from the outside internet - wireless cards may be disabled or even removed
- Data is required to be transferred physically, for example on a USB drive

Resilience

- Regular backups
- Standby servers
- 'Availability Zones' within Cloud Infrastructure

Regions
Coming Soon

https://aws.amazon.com/about-aws/global-infrastructure/

Privacy & Anonymity

- Use a VPN!
- Rotating proxy services are available from many providers, and can simulate 'residential IPs' for use in general browsing and for automated tasks like web scraping on a large scale

SHEFFIELD Ethical Student Hackers
Breaking into security.

Burp Suite

What is a proxy?

- Essentially a ‘middle-man’ for network traffic - it’s an extra step between source & destination

What is Burp?

- A powerful proxy that captures your HTTP/HTTPS traffic and allows you to modify it on the fly
- It’s great for analysing how a website/server functions and crafting attack payloads

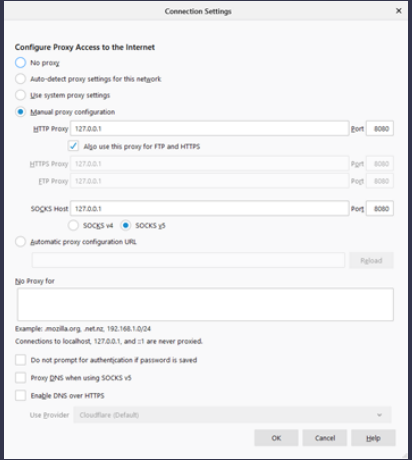
What can it do?

- Show you all the HTTP traffic passing through, and info about data, headers, and more
- Intercept a request before it is sent, and allow modification of details such as data and URLs
- Send a request to the ‘Repeater’ to allow fine-grained control of its properties, and define an attack payload in the ‘Intruder’ for fuzzing and automation
- Encode/decode with common web encodings such as Base 64

Setting up Burp

Install here: https://portswigger.net/burp/communitydownload

Configure your browser to proxy your HTTP traffic



Connection Settings

Configure Proxy Access to the Internet

- ( ) No proxy
- ( ) Auto-detect proxy settings for this network
- ( ) Use system proxy settings
- [✓ Manual proxy configuration]
  - HTTP Proxy: 127.0.0.1 Port: 8080
  - [✓] Also use this proxy for FTP and HTTPS
  - HTTPS Proxy: 127.0.0.1 Port: 8080
  - FTP Proxy: 127.0.0.1 Port: 8080
  - SOCKS Host: 127.0.0.1 Port: 8080
  - SOCKS v4 [✓ SOCKS v5]
- ( ) Automatic proxy configuration URL

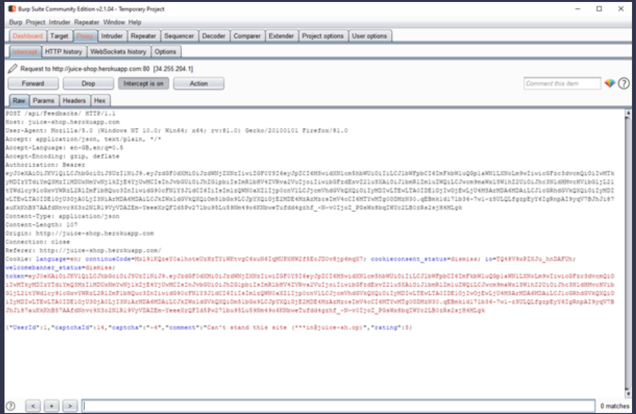
No Proxy for

Example: .mozilla.org, .net.nz, 192.168.1.0/24
Connections to localhost, 127.0.0.1, and ::1 are never proxied.

- [ ] Do not prompt for authentication if password is saved
- [ ] Proxy DNS when using SOCKS v5
- [ ] Enable DNS over HTTPS

Use Provider: Cloudflare (Default)

Go to the 'Proxy' tab, turn on Intercept, and watch the traffic flow!



Burp Suite Community Edition v2.1.04 - Temporary Project
Burp Project Intruder Repeater Window Help
Dashboard Target Proxy Intruder Repeater Sequencer Decoder Comparer Extender Project options User options
Intercept HTTP history WebSockets history Options
Forward Drop Intercept is on Action
Comment this item
Raw Params Headers Hex
0 matches

Request to http://juice-shop.herokuapp.com:80 [34.255.204.1]

```
POST /api/Feedbacks/ HTTP/1.1
Host: juice-shop.herokuapp.com
User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:81.0) Gecko/20100101 Firefox/81.0
Accept: application/json, text/plain, */*
Accept-Language: en-GB,en;q=0.5
Accept-Encoding: gzip, deflate
Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiOiIwMTk
yMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiJ1bmRlZmluZWQiLCJwcm9maWxlSW1hZ2UiOiJhc3NldHMvcHVibGljL21
tYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIwLTEwLTA0IDE1OjIwOjEwLjU4MSArMDA6MDAiLCJ1cGRhdGVkQXQiOiIyMDI
wLTEwLTA0IDE1OjU3OjA0LjI3NiArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2MDE4MzAzMzcsImV4cCI6MTYwMTg0ODMzN30.qEBmk1d171b36-7w1-z5ULQLfgzpEyY6IgRnpAI9yqV7BJhJi87
auKkKhB57AAfdNnvc9X3o2N1R19VyVDAZEm-YeeeXrQFId5Pw27lbu95Lu59Nm49o6X3bweTufdd4gzhf_-N-v0IjoZ_PGsMx8bqIWYc2LB0zRs2ajH4MLgk
Content-Type: application/json
Content-Length: 107
Origin: http://juice-shop.herokuapp.com
Connection: close
Referer: http://juice-shop.herokuapp.com/
Cookie: language=en; continueCode=Mxl91KQzeY0alhoteUzKzTYiMKtvgC4xuN6IqMUPXNM2f5EoJD0v8jp4mqX7; cookieconsent_status=dismiss; io=TQ48V9xPZXJu_hnDAFUh;
welcomebanner_status=dismiss;
token=eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJzdGF0dXMiOiJzdWNjZXNzIiwiZGF0YSI6eyJpZCI6MSwidXNlcm5hbWUiOiIiLCJlbWFpbCI6ImFkbWluQGp1aWNlLXNoLm9wIiwicGFzc3dvcmQiO
iIwMTkyMDIzYTdiYmQ3MzI1MDUxNmYwNjlkZjE4YjUwMCIsInJvbGUiOiJhZG1pbiIsImRlbHV4ZVRva2VuIjoiIiwibGFzdExvZ2luSXAiOiJ1bmRlZmluZWQiLCJwcm9maWxlSW1hZ2UiOiJhc3NldHMvcHVib
GljL2ltYWdlcy91cGxvYWRzL2RlZmF1bHQuc3ZnIiwidG90cFNlY3JldCI6IiIsImlzQWN0aXZlIjp0cnVlLCJjcmVhdGVkQXQiOiIyMDIwLTEwLTA0IDE1OjIwOjEwLjU4MSArMDA6MDAiLCJ1cGRhdGVkQXQiO
iIyMDIwLTEwLTA0IDE1OjU3OjA0LjI3NiArMDA6MDAiLCJkZWxldGVkQXQiOm51bGx9LCJpYXQiOjE2MDE4MzAzMzcsImV4cCI6MTYwMTg0ODMzN30.qEBmk1d171b36-7w1-z5ULQLfgzpEyY6IgRnpAI9yqV7B
JhJi87auKkKhB57AAfdNnvc9X3o2N1R19VyVDAZEm-YeeeXrQFId5Pw27lbu95Lu59Nm49o6X3bweTufdd4gzhf_-N-v0IjoZ_PGsMx8bqIWYc2LB0zRs2ajH4MLgk

{"UserId":1,"captchaId":14,"captcha":"-4","comment":"Can't stand this site (***in@juice-sh.op)","rating":5}
```

Note: You can visit localhost:8080 in your browser to import Burp's certificate and prevent those annoying MITM warnings

SHEFFIELD Ethical Student Hackers
Breaking into security.

Using Burp

Proxy

- 'Intercept' determines whether traffic is held up before being passed on
- 'HTTP History' shows all requests and responses captured by Burp
- From here, you can send requests to the Repeater/Intruder for more options
- You can also tell Burp not to intercept certain requests using the 'Action' Button

Repeater

- Send a request here using Ctrl+R!
- Once it's in the Repeater, you can modify anything you like and resend

Intruder

- You can configure 'positions' and 'payloads' here to set up an automated attack
- See our worksheet from Session 3 for a detailed look at using the Intruder!

Upcoming Sessions

What's up next?
www.shefesh.com/sessions

7th-8th November - HackSheffield 6!
(see https://hacksheffield.com/)

9th November - All the Shells!

16th November - Enumeration

23rd November - Privilege Escalation

30th November - Open Source Intelligence

Any Questions?

www.shefesh.com

Thanks for coming!