Student number:	2467273
Course title:	COMPSCI4062 Cyber Security Fundamentals (H)
Questions answered:	ALL

1.

a) The capitalised text (SGMNYCWEVT) is encrypted with two different ciphers, the decryption of which is described in the note. ~~Firstly, it needs to be decrypted using Caesar's shift/cipher because the text's encryption is described as a shift to the right of the letters of the English alphabet (A-Z). Thus, it needs to be decrypted as a shift to the left: QEKLWAUCTR. Secondly, the Vigenere cipher is used with "pass" as the keyword, meaning that the previously decrypted text needs to be decrypted in the Vigenere table with this keyword, which, when decrypted, shows: BESTHACKER~~.

[Above text was written before MS Teams announcement; the following is after.] Since the encryption step 1 was Caesar's shift, which shifts the letters to the right in the English alphabet (A-Z) and step 2 was to encrypt the product of Caesar's shift with a Vigenere cipher with "pass" as the keyword using the Vigenere table, these steps need to be reversed to acquire the decrypted text. Firstly, decrypting using the Vigenere cipher with keyword "pass" gets "DGUVJCEMGT" as the product. Secondly, taking this and decrypting it by shifting each letter 2 positions to the left (reverse of encryption) according to Caesar's shift, we get: BESTHACKER.

b) If we transform ASECRETMESSAGE in different ways:

- Using the method in part a) to **encode** the string in the same way: RUYWIGNGVUMUXG
- Using the method in part a) to **decode** the string in the same way: JQKIACZSNQYGPC
- Using substitution of letters in the original for special characters/numbers: @S3CR3TM3ss4G3

Both of the first possible passwords (which are substitutions of characters by way of encryption/decryption) are not very strong passwords because, while they do not hold any meaningful text in their raw form (which is another way a password could be more easily brute forced with a dictionary attack, thus making the passwords a little better than, e.g., "MYCATMITTENS"), they are only collections of same-case letters, which is a very easily breakable format by brute force password-guessing algorithms, which prioritise same-case letters first. The third password would be much harder to break (and is, therefore, more secure than the first two) because it involves special characters (@), differently capitalised letters (S and s), and digits intermixed all together, all of which add a level of complexity to the password, each new type of character exponentially adding to the potential passwords a brute-force algorithm needs to check.

2.

a) White box penetration testing would be much more viable to be done in 2 weeks than black box testing, which takes longer (20-30 days), because there is much more information available to pen testers in a white box environment than in a black box one. With this information, pen testers can compare the security measures of the website with standards used in other platforms and applications, which can then be used to find possible vulnerabilities not being patched. This testing is not very realistic since it does not check how the company would respond to an attack, but it does allow more thorough testing of a system since access can be given to the testers to also test core structures, not just what can be accessed through alternative/penetrative methods.

b) From Figure 1:

- The Operating System (OS) for the first host is Apache/2.4.7 Ubuntu. This information is very useful to an attacker because knowing the OS allows them to find out the implicit vulnerabilities in the OS itself, which might not have been patched and are exploitable since OS vulnerabilities can be used to gain access to the system itself as it uses that OS.
- IPv4 address of the first host is 45.33.32.156. With the knowledge of IP, the attackers can find out other information by doing online lookups, e.g., what other IPs it communicates with, and what the hardware structure of the IP address location is. An attacker might find vulnerabilities in the hardware and/or in the systems that the other IP addresses belong to, all of which could propagate to the original host.
- The language used client-side of the application server is JavaScript. This would allow and attacker to try out and possibly find potential cross-site scripts (XSS) that could be run on the application to gain information about the underlying technologies of the application or, for example, gain cookie information from all other users of the application, which could be a privacy catastrophe since a lot of vital/sensitive information is sometimes stored in cookies.

From Figure 2 (DNS Record):

- The IPv6 address of one of the hosts is 2600:3c01::f03c:91ff:fe18:bb2f. This is exploitable in the same way that an IPv4 address is, described under Figure 1, but it might reveal more information specific to the host since IPv6 addresses are much more unique than IPv4 ones.
- Mail exchange information in last 3 rows (with “MX” in second column). This can be used to spoof the services used in emails for the host and possibly send seemingly realistic phishing emails, thus attacking the people using the host IP address, or possibly to find vulnerabilities in the services themselves and intercept information transmitted through them.
- Information about sample output for TXT input (with “TXT” in second column). This can also be used to spoof information regarding all of the addresses used in order to create and send phishing emails, but it would also be possible to find out more about the vulnerabilities present in the IP addresses listed in the output.

c) Who-Is search tools would be perfect for gaining this sort of information from a given IP address, for example, www.ipvoid.com could be used (with Whois Lookup) to find information similar to that of Figure 1 (or, with DNS Record searches, to find DNS Records information like in Figure 2). Additionally, https://centralops.net could be used to find similar information, but possibly even more. Other tools, like Netcraft and command-line commands, like “ping” and “tracert”, can be used

to find similar information, the latter of which would show the IP addresses accessed during communication with a target address, the duration of communication between them and other valuable information.

d) This output includes additional information not seen in figures 1 & 2, like open ports and what service they use, as well as network distance, and also a confidence rating of what OSs the host could be running on, starting with most likely to least likely since there are not absolutely exact matches found. This is probably output from nmap (/**Zenmap**). It also shows whether the host is running ("up"), what its latency is and additional service info, as well as how many ports on the system are closed. For each of the open ports, more information is given on what version and OS they are running on.

e) Once all this information is acquired through reconnaissance and scanning, attackers might gain access to the system by exploiting vulnerabilities found in any of these pieces of information found. For example, Metasploit could be used to gain this access and then maintain access by building backdoors with Trojan horses. With this continuous access, information could be monitored either on the screen or keys tapped on the keyboard (with a keylogger) and then passed onto the attacker.

Alternatively, a tool like Armitage could be used to gain and maintain access, which then could be used to introduce viruses into the system, like ransomware, which would encrypt all of the information of the victim and ask for money or certain access privileges from the owner in order to decrypt their information and get it back.

3.

a) Pieces of information:

- There are two communicating IP addresses, 10.1.1.101 and 10.1.1.1, which seems to be handshaking from row 233 to row 242, since they are exchanging SYN and ACK packets and send a finishing acknowledgement in row 242.
- Two JPEG pictures seem to be accessed through HTTP with flag 200 OK, which signals that the information was successfully accessed.
- There are a lot of ACK packets sent between the HTTP communication, which seems to indicate a large file being sent during the communication since there are no SYN packets sent at that point, but many ACK packets sent during a short duration (about 0.2 milliseconds).
- There are many rows containing the phrase “TCP segment of a reassembled PDU”, which talks about the Transmission Control Protocol controlling communication by splitting up a large Protocol Data Unit and then reassembling it because of efficiency gained by packets being sent in this way to get access to larger storage pictures.
- The ports that seem to communicate with each other are 80 (the default HTTP port), 3197, 3198, and 3199 (last three of which are personally employed ports). The port information can be used for attackers to understand which ports are busy and which, possibly important ones, are not.
- The protocol used for this communication is HTTP without any mention of TLS (Transport Layer Security), which means that the information transmitted through this protocol is not encrypted in any way, like it would be if HTTPS was used.

b) One of these scenarios could be information access of data being transmitted, which could easily be done and gain understandable/interpretable information since it is not encrypted in any way, just the basic HTTP is used. This could be done by a Man-in-the-Middle (MitM) attack, where the attacker could monitor the traffic and would not even need to bother with any encryption/decryption attacks since the data is not encrypted in the first place.

A different scenario could, based on this knowledge, either send frequent requests for these large images or just send many SYN packets to the host’s port, which would cause a SYN Flood, a type of Denial of Service (DoS) attack, to the system and cause it to malfunction/crash. All of the ports encountered in this Wireshark sample could be exploited and flooded with SYN packets for a more effective flood.