**CSC4222 Fundamentals of Cybersecurity**

**Homework 4**

**Part I: Answer the following questions**

1. Application Layer Security

Compared to security at the network and transport layers, what unique security challenges are encountered at the application layer? Please illustrate your explanation with concrete scenarios or protocols.

Challenges include review filtering user inputs, user authentication/access checks, creating/protecting cookie sessions, secure apis. In comparison from security at network and transport layers as compared to the application level, applications are aware of user permissions and roles while the network levels don’t care about the user identity as it intercepts the packets, eavesdrop, or causes denial of services havoc.

At the application layer, we need to review the user inputs so that it doesn’t inject any malicious code such as an SQL injection attack which would allow the attack to bypass credentials, steal/encrypt data or delete the entire table.

Here is an example, lets say David buys an item from megastore:

GET https://megashop.com/orders/20001

An attacker can change do

Get https://megashop.com/orders/12345

And get David’s sensitive information related to the order, which can be patched in the application layer of the buy order checks the user id compared to the owner id.

1. Automated Security Policies and Network Security Management

Automated security policies (such as SDN-based automatic traffic isolation) can respond swiftly when an attack occurs. Please explore how to design an automated security management system, explaining its workflow, key technologies, and the potential challenges encountered during implementation from the perspective of network layering.

*SDN (Software Defined Networking) is an approach to network architecture that separates the control plane from the data plane, allowing for centralized management and dynamic configuration of network resources.*

An automated security management system designed revolved Monitor network activity(Observe incoming packets and traffic flow sent through network), Detect threats(find any anomalies from normal traffic flow or compare signatures for known attacks), Deciding security responses(decide the actions in reference to the polies either its warning or blocking), and logging(keep records of all actions taken for review). Key technologies used include using Wireshark, Machine Learning anomaly models or AI for detection and decision making, Openflow, or Splunk. Challenges include integration and false positives, it provides unmatched agility, visibility, and reaction time compared to traditional static defenses.

1. Exploration of the Security of Remote Work and VPN Technology

With the growing prevalence of remote work, VPNs have become a vital means of securing data transmission. In conjunction with our classroom discussions on IPsec, please research the principles by which VPNs implement encryption and tunneling at different network layers, and analyze the potential security risks and improvements in the face of man-in-the-middle attacks.

VPN creates a secure tunnel for the user and the network that encrypts the data set through the network to prevent eavesdropping. In the network Layers, in layer 2, they apply Layer 2 Tunneling Protocol which encrypts the Entire Ethernet frame. In layer 3 Network Layer, it transports and encrypts the IP packets payload and uses algorithms learned in class such as AES, SHA, DES. Then in the 4th to 7th layer, it is where security measures like the SSL VPN, HTTPS Proxy apply TLS/SSL to encrypt Application-specific data and web traffic. This is why they meet and apply the CIA principals by using Encapsulation, Protocols, and Interface

The potential security for using VPNs though however is that there are fake VPNs that exist that tricks user into connecting to a server that steals their data, weak authentication, and does not protect against Malware.

You can improve your security against the man in the middle attacks from VPNs by using multi factor Authentication and use trusted certified VPNs.

1. Explain the concept of the Same-Origin Policy (SOP). Why is it important in web security?

The Same-Origin Policy (SOP) is something in web browsers that restricts how documents or scripts from one location/endpoint interact with resources from another origin. What makes up this original is the Scheme: Https/Http, Host:.com .org directories, and port number. This protects users from accessing malicious websites that tries to steal or access sensitive data without permission such as cookies and session tokens.

1. Web Application Firewalls (WAFs) Analysis:

In our class, we learnt knowledge about firewall. But WAF plays different roles compare to traditional firewall. WAF is designed to secure web applications from various attacks. Try to find two kind of WAF using google/bing and discussing their strengths, weaknesses, and the types of attacks they are best suited to defend against.

Then compare the two WAFs with our traditional firewall we learnt in class.

WAF:

Network-based WAF is generally hardware-based.

The benefits it provides is its high performance due to low latency since it is installed locally in the hardware, it is easier to manage the policies and rules, great traffic inspections. Ease of integrating into network security applications and doesn’t require much change in backend code. However, its cons include being one of the most expensive option, require the storage and maintenance of physical equipment, can accidentally block legit traffic as false positives, and struggles with protecting API applications.

A host-based WAF is integrated into the application’s software.

Benefits of this include being less expensive, easier to customize, inspect traffic in context of session state and file uploads, and isn’t reliant on network.

Cons include consuming a lot of resources from CPU, memory, and I/O, poses a security risk if it gets compromised accessing the machine of the application, hard to update policies and rules.

When comparing Host based vs Network based WAF, They differ in that Host based is less expensive than Network based WAF, Host based WAF has more Api control but harder to update its rules and policies while Network based WAF is easier to update the rules but struggle with protecting api, Network based WAF requires physical maintenance while Host based consumes the hardware’s resources. Both WAFs have scalability issues though as Network based require more environmental room to install while Host based have issues with managing large servers.

In comparison to our traditional firewalls, traditional firewalls are impenetrable unlike Host-based which has security risks, in addition, the wafs operate at layer 7 while our traditional firewall operates at layer 3 and 4. In addition, traditional firewalls block external threats but doesn’t check for internal threats such as if machine is compromised, it won’t check the internal while both WAFs check what’s being transported.

1. Describe a scenario where network flow analysis could be crucial in solving a cybercrime. What specific data would you look for?

Network flow analysis can help us identify abnormalities or patterns in network traffic, so for example, if a laptop or machine within a company was compromised and is consuming a lot of resources for malicious intent. Specific datas to look for includes the bytes that was sent/receives, timestamps, source and destination, protocols, and connection time.

1. Describe how HTTPS works to secure data transmission between a client and a server. What are the key components involved in establishing an HTTPS connection?

HTTPS is a secures version of HTTP that encrypts data transmission between the client and the remote server by using TLS to ensure CIA is followed. Key components of HTTPS include data encryption via TLS, Certificate checks to authorize safe and legitimacy of digital signatures, and Handshake protocol between client and server of authentication, negotiation encryption, and key sharing

1. What is Cross-Origin Resource Sharing (CORS), and how does it relate to the Same-Origin Policy?

CORS is a security measure used by web browsers to control resource sharing such as APIs. The same origin policy was what was enforced before CORS which only allowed the web to make requests to the same web page. CORS is a lenient version of the Same-Origin policy that allows for web to make requests to different origins.

**Part II: Analyzing Port Scanning Behavior in Captured Network Traffic**

You have received a PCAP file (attack.pcapng) that captured simulated attack traffic. Please use Wireshark or another network analysis tool to perform a detailed analysis of this file and complete the following tasks:

**Attacker Identification:**

* Analyze the packets to identify the IP address of the attacker who initiated the scanning behavior.  
  *Hint:* Look for the source IP sending a large number of probe packets (such as TCP SYN packets or UDP packets). You may use a filter like tcp.flags.syn==1 && tcp.flags.ack==0 (for a TCP SYN scan) for preliminary filtering.

Attack ip address is most likely due to the high amount of outgoing packets they sent 172.16.1.101 however the ip address 172.16.1.4 is also very active which could be a potential attack as well.

**Target Host Identification:**

* List all the IP addresses of the target hosts that were scanned in the packets.  
  *Hint:* Focus on the bidirectional traffic between the attacker and the target hosts. You might need to use a filter such as ip.addr==<target IP> for further verification.
* Target Host:
* 172.16.1.1
* 172.16.1.4
* 172.16.1.10
* 172.16.1.100
* 172.16.1.102
* 172.16.1.255

**Port Scanning Analysis:**

* For each target host, count and list the scanned port numbers.
* Explain the common services associated with these ports (for example: 80 for HTTP, 443 for HTTPS, 22 for SSH, etc.), and discuss the possible scanning intent of the attacker.

Target IP Ports Scanned

172.16.1.1 67 (UDP)

HTTP Web server testing, checks for web services

172.16.1.4 Full TCP scan ranging from port 1 to 65389

Performs a full TCP port scan which includes HTTPS,HTTP,MySQL, etc, possibly intending to scan for vulnerabilities

172.16.1.10 80, 1032

Common services used include HTTP for web services and 1032 for Ephemeral, which isn’t a well-known service, but online resources and documentation suggest it is a port for trojans or worms suggesting intent to target a vulnerability with a malware.

HTTP, multiple high ports used for backdoors and web scan

172.16.1.100 139

Port 139 is NetBIOS Session Service (SMB), possible attack intent is to check if it is windows host for potential targets for exploit

172.16.1.102 80,21,24,3389,445,5007

Common services include FTP servers probably intending to attempt unauthorized access, HTTP, RDP which is for targeting remote desktop services for exploitation, SMB which is mainly used for file sharing so maybe intercepting files, and 5007 is just media streaming services.

172.16.1.255 137,138(UDP)

Both ports are mainly for NetBIOS services which is mainly network service so the attacker may intend to find other hosts in the local network before targeting them

**Scanning Technique Discussion:**

* Based on the captured traffic, determine the type of scan used by the attacker (e.g., TCP SYN scan, ACK scan, UDP scan, etc.), and support your conclusion with specific packet characteristics.
* In your report, describe the filters and analysis steps you used, and explain the key criteria for differentiating between the various scanning techniques.

The attacker used a TCP SYN scan to attack various ports across the network. This is evidenced by the fact that there are RST, ACK responses (tcp.flags.ack == 1 and tcp.flags.syn == 0) from target systems like 172.16.1.4. This indicates that the attack sent SYN packets and the target ports replied RST, ACK because it was a closed port receiving SYN so no full handshake was established but the attacker did check the state of the ports.

**Part III: Content Security Policies**

Recall that content security policy (CSP) is an HTTP header sent by a web site to the browser that tells the browser what it should and should not do as it is processing the content. The purpose of this question is to explore a number of CSP directives. Please use the [CSP specification](https://developer.mozilla.org/en-US/docs/Web/HTTP/Headers/Content-Security-Policy) to look up the definition of the directives in the questions below.

1. Explain what the following CSP header does:

Content-Security-Policy: script-src 'self'

What is the purpose of this CSP directive? What attack is it intended to prevent?

CSP directives stops other sources of scripts to be injected and executed on another web page, and this script loads only the script from the origin, to reduce the risk of script-based attacks by enforcing that scripts only come from trusted same-origin source

It in intended to prevent Cross-Site Scripting attacks.

1. What does the following CSP header do:

Content-Security-Policy: frame-ancestors 'none'

What attack does it prevent?

The head does not allow any domain, even if it is coming from origin to be used in the frame which prevents another website using their website on theirs to frame a legit website in order to redirect them to something else that wasn’t intended. This is to prevent clickjacking attacks.

1. What does the following CSP header do:

Content-Security-Policy: sandbox '[allow-scripts](https://developer.mozilla.org/en-US/docs/Web/HTML/Element/iframe#attr-sandbox)'

This script applies the sandbox policy which restricts what the page can do, kind of like putting software in quarantine in windows. It only allows JavaScript but everything else is restricted.

Suppose a page loaded from the domain www.xyz.com has the sandbox CSP header, as above. This causes the page to be treated as being from a special origin that always fails the same-origin policy, among other restrictions. How does this impact the page's ability to read cookies belonging to www.xyz.com using Javascript? Give an example where a web site might want to use this CSP header.

The website [www.xyz.com](http://www.xyz.com) can run scripts but it cannot access cookies or any other origin data from JavaScript which will result in the scripts returning nothing because it has no input. A web site might want to use this in an iframe to quarantine embed untrusted content on their website to prevent cookies from being stolen.

**Part III: A Dive into Ethereum and Smart Contracts**

**Exploring Ethereum's Capabilities Beyond Digital Currency:**

Ethereum is often hailed as a revolutionary technology that extends blockchain's utility beyond just a digital currency platform, as exemplified by Bitcoin, into a more expansive ecosystem for decentralized applications (DApps). Your task is to delve into Ethereum's underlying architecture and its native programming language, Solidity. Investigate how Ethereum's smart contract functionality differentiates it from Bitcoin and how this has contributed to its adoption in various sectors.

1. **Describe the foundational principles of Ethereum and how it expands upon the basic concepts of blockchain technology introduced by Bitcoin.** Focus on Ethereum's virtual machine (EVM), the role of Ether (ETH) within the network, and the significance of smart contracts.

Ethereum is basically a virtual machine that allows for execution of smart contracts and decentralized applications called Dapps. Bitcoin was a decentralized digital currency, and it is a scripting language that was intentionally limited functionality to allow for security and simplicity but with Ethereum, it allows users to build and deploy Dapp going pass the rules defined by the smart contract anonymously which resulted in trust issues since smart contracts can automatically execute and enforce rules without human intervention.

1. **Analyze a real-world application of Ethereum in a sector other than finance, such as supply chain management, voting systems, or digital identity verification.** Discuss how Ethereum's features are leveraged in this application, the benefits it brings over traditional systems, and any challenges or limitations encountered.

Traditional systems rely on centralized entities that store personal data, making them prone to breaches. Ethereum introduces a self-sovereign identity model, allowing individuals to control their personal information. Users can create digital identities secured by Ethereum’s blockchain, which they can selectively share with trusted parties. This reduces reliance on intermediaries and ensures privacy. For example, Ethereum-based platforms like uPort enable verifiable credentials for accessing services like voting, healthcare, and finance without exposing sensitive data. The government in Zug in Switzerland who is also called Crypt Valley allowed citizens to register on Ethereum to receive a uPort identity to log into government services for blockchain voting. The challenge however is that it isn’t very user experience friendly as managing private keys and whatnot is difficult for the individual, and frequent smart contracts can be expensive.

**Smart Contracts: The Building Blocks of Decentralized Applications:**

Smart contracts are self-executing contracts with the terms of the agreement directly written into lines of code. These digital contracts run on blockchain platforms, like Ethereum, and are immutable and distributable, making them a key component in a wide range of decentralized applications (DApps).

1. **Elucidate the concept of smart contracts, detailing their characteristics, how they operate within blockchain ecosystems like Ethereum, and the advantages they offer over traditional contract law.**

**Smart contract’s code cannot be changed to ensure trust, it can operate independent of user interface, and it’s secured by mechanism of blockchains. Smart contracts are written in Solidity or Vype that is deployed on the Ethereum blockchain from a transaction. It then runs on the Ethereum Virtual Machine and consumes gas which is a fee for executing transactions and smart contracts on a blockchain, which prompt Dapps to interact with the smart contracts through web interfaces. It’s advantages over traditional contract laws include not needing to trust the other party, only the code, smart contracts can’t be changes to avoid malicious intent, and contract is publicly accessible to be verified.**

1. **Select a novel application of smart contracts outside the commonly cited examples (e.g., avoiding finance-related applications like ICOs or DeFi).** Provide a comprehensive overview of this application, including how smart contracts are employed, the specific problems they address, and any potential issues or criticisms associated with their use in this context.

Smart contracts can help musicians get instantly paid without having to rely on a middleman since the smart contract runs automatically once the predetermined rules are met. For example, the music app Spotify takes months to pay musicians due to the fact that the money goes through multiple middlemen before it arrives to the musician. However, when smart contracts are employed, the musician can get instantly paid without having to go through multiple middleman which saves both time and the requirement of having to trust the middleman. However, since smart contracts are still relatively new, legal systems will revert back to traditional contract laws if a dispute arises and it cost gas fee to run smart contracts.

***Submssion***:

Submission:

• Upload an electronic copy (MS word or pdf) of your answer sheet.

• Please add the homework number and your name at the top of your answer sheet.

• Please write down your answers with the question number only in the answer sheet.

• Name your file in the format of HW3\_FisrtnameLastname (eg.

HW3\_PengWang.docx, HW3\_PengWang.pdf)