TASKS

AI FOR CYBER SECURITY

Sanjaykrishnaa R S

21BRS1283

CSE with AI and Robotics

Chennai campus

23/08/23

Task: write a brief description for the top 10 most notorious hackers in the world

1-kevin Mitnick- is top most hacker he is not a active hacker he worked in his late 20s,Mitnick was known for his high level of technical expertise in computer systems and telecommunications. He was skilled in social engineering techniques, which involve manipulating people into divulging confidential information.

He fits into black hat hacker category but Although Mitnick ultimately went white hat, he may be part of the both-hats grey area.

2-Anonymous-they refers to a hidden hierarchy they worked as groups and they began in 2003 on 4chan message boards. they groups form little organization and major focus is on

social justice they are impossible to catch up due to lack of their real hierarchy. They are not defined into any hat category they can be called as hacktivism or if we must categorize them, we can also call them grey hat hacker as they do both

3-Adrian lamo-Lamo was often referred to as a "grey hat" hacker due to his actions falling

between the realms of black hat (malicious hacking) and white hat (ethical hacking).in 2001 a 20 year old adrian hacked yahoo and add his own quote. One of Lamo's most notable actions was his involvement in the arrest of Chelsea Manning (formerly known as Bradley Manning), a United States Army intelligence analyst who leaked classified documents to WikiLeaks. Manning contacted Lamo in 2010 and confided in him about the leaks he passed away at age of 37

4-Albert gonzalez-was a notorious hacker who was involved in several high-profile cybercrime activities. He is often associated with the largest credit card data breaches in history.

Gonzalez led a group of hackers responsible for stealing credit card data from major retailers, including TJX Companies (which owns TJ Maxx and Marshalls) and Heartland Payment Systems.

These breaches resulted in the theft of tens of millions of credit card numbers and caused significant financial losses for individuals and companies.

Albert Gonzalez is generally classified as a "black hat" hacker due to his involvement in malicious and illegal cyber activities, specifically cybercrimes such as credit card data theft and subsequent fraud.

5-Matthew Bevan and Richard Pryce-Matthew Bevan and Richard Pryce are a team of

British hackers who hacked into multiple military networks in 1996, including Griffiss Air Force Base, the Defense Information System Agency and the Korean Atomic Research Institute (KARI). Bevan (Kuji) and Pryce (Datastream Cowboy) have been accused of nearly starting a third world war after they dumped KARI research onto American military systems. They demonstrated that even militiary networks are vulnerable.

The hacking activities attributed to Bevan and Pryce were not solely malicious or criminal in nature, as their actions were sometimes driven by curiosity and the desire to explore and understand computer systems. However, their actions also involved unauthorized access to computer systems and networks, which is a characteristic of black hat hacking.

The "gray hat" classification acknowledges that their motivations and activities might not fit neatly into the white or black hat categories. They were neither purely

malicious nor strictly ethical hackers. Instead, their actions had shades of both, making their classification complex and open to interpretation.

6-Jeanson James Ancheta

Ancheta was involved in creating and controlling a network of compromised computers, known as a "botnet," for financial gain.Jeanson James Ancheta's activities highlight the criminal exploitation of compromised computers for financial gain. His case underscores the importance of cybersecurity measures to prevent the creation and control of botnets, as well as the legal consequences that cybercriminals can face when caught.

Jeanson James Ancheta is typically classified as a "black hat" hacker due to his involvement in various cybercriminal activities that involved unauthorized access to computers, creation and control of botnets, and distribution of malware for financial gain.

7-Michael Calce

Michael Calce, also known by his online handle "Mafiaboy," is a Canadian hacker who gained notoriety for launching a series of high-profile distributed denial-of-service (DDoS) attacks in the late 1990s. These attacks targeted several prominent websites, including major e-commerce and online platforms.Mafiaboy's most notorious attack occurred in 2000 when he launched a DDoS attack against several high-profile websites, including Yahoo!, Amazon, eBay, and CNN. These attacks overwhelmed the targeted websites' servers with a flood of traffic, causing them to become inaccessible to users.

Mafiaboy's activities fall under the category of "black hat" hacking, as he engaged in malicious activities that disrupted online services, caused financial losses to businesses, and impacted the experiences of countless internet users.

8-Kevin Poulsen- also known by his online handle "Dark Dante," is a former black hat hacker who gained notoriety in the 1980s and 1990s for his hacking activities. He was involved in several hacking incidents, one of which was particularly significant and led to his arrest.

Poulsen's most notable action was hacking into the telephone lines of a Los Angeles radio station, KIIS-FM, in 1990. He manipulated the station's phone system to ensure that he would be the 102nd caller in a contest, winning a Porsche 944 S2.

Poulsen's transition from black hat hacking to ethical cybersecurity work makes his story unique. He is often seen as an example of someone who managed to transform their skills and experiences for positive contributions to the field.

9-Jonathan James-, also known by his online handle "comrade," was a hacker known for his involvement in various cybercriminal activities during the late 1990s and early 2000s. He gained attention for being one of the youngest hackers to be arrested for cybercrimes.NASA hack.

at that time he was identified as a 58-year-old Greek mathematician. Reportedly, he had been hacking into the Dassault Group, for almost half a decade. During that time, he stole cutting edge weapons technology software and data which he then sold to 250 individuals around the world.

It can classified under black hat as it uses illegal activity for weapon sell.

His actions are generally categorized as "black hat" hacking due to their malicious nature and the legal ramifications they carried. His case also underscores the importance of addressing mental health issues and providing support to individuals, especially young ones, who may be drawn into cybercriminal activities.

10-ASTRA-his hacker differs from the others on this list in that he has never been publicly identified. However, according to THE daily mail some information has been released about ASTRA. Namely that he was apprehended by authorities in 2008, and

at that time he was identified as a 58-year-old Greek mathematician. Reportedly, he had been hacking into the Dassault Group, for almost half a decade. During that time, he stole cutting edge weapons technology software and data which he then sold to 250 individuals around the world.

It can be classified under black hat as it uses illegal activity for weapon sell.

24/08/23

Task: List some common ports and their vulnerabilities

Here are the most vulnerable ports regularly used in attacks:

Ports 20 and 21 (FTP)

Port 20 and (mainly) port 21 are File Transfer Protocol (FTP) ports that let users send and receive files from servers.

FTP is known for being outdated and insecure. As such, attackers frequently exploit it through:

- Brute-forcing passwords
- Anonymous authentication (it’s possible to log into the FTP port with “anonymous” as the username and password)

- Cross-site scripting
- Directory traversal attacks

Port 22 (SSH)

Port 22 is for Secure Shell (SSH). It's a TCP port for ensuring secure access to servers. Hackers can exploit port 22 by using leaked SSH keys or brute-forcing credentials.

Port 23 (Telnet)

Port 23 is a TCP protocol that connects users to remote computers. For the most part, Telnet has been superseded by SSH, but it's still used by some websites. Since it's outdated and insecure, it's vulnerable to many attacks, including credential brute-forcing, spoofing and credential sniffing.

Port 25 (SMTP)

Port 25 is a Simple Mail Transfer Protocol (SMTP) port for receiving and sending emails. Without proper configuration and protection, this TCP port is vulnerable to spoofing and spamming.

Port 53 (DNS)

Port 53 is for Domain Name System (DNS). It's a UDP and TCP port for queries and transfers, respectively. This port is particularly vulnerable to DDoS attacks.

Ports 137 and 139 (NetBIOS over TCP) and 445 (SMB)

Server Message Block (SMB) uses port 445 directly and ports 137 and 139 indirectly. Cybercriminals can exploit these ports through:

- Using the EternalBlue exploit, which takes advantage of SMBv1 vulnerabilities in older versions of Microsoft computers (hackers used EternalBlue on the SMB port to spread WannaCry ransomware in 2017)
- Capturing NTLM hashes
- Brute-forcing SMB login credentials

Ports 80, 443, 8080 and 8443 (HTTP and HTTPS)

HTTP and HTTPS are the hottest protocols on the internet, so they're often targeted by attackers. They're especially vulnerable to cross-site scripting, SQL injections, cross-site request forgeries and DDoS attacks.

Ports 1433,1434 and 3306 (Used by Databases)

These are the default ports for SQL Server and MySQL. They are used to distribute malware or are directly attacked in DDoS scenarios. Quite often, attackers probe these ports to find unprotected database with exploitable default configurations.

Port 3389 (Remote Desktop)

This port is used in conjunction with various vulnerabilities in remote desktop protocols and to probe for leaked or weak user authentication. Remote desktop vulnerabilities are currently the most-used attack type; one example is the BlueKeep vulnerability.

28/08/23

Task: Explain any 10 web attacks other than the top 10 owasp vulnerabilities

1) Binary Planting

Description

Binary planting is a general term for an attack where the attacker places (i.e., plants) a binary file containing malicious code to a local or remote file system in order for a vulnerable application to load and execute it.

There are various ways this attack can occur:

Insecure access permissions on a local directory allow a local attacker to plant the malicious binary in a trusted location. (A typical example is an application installer not properly configuring permissions on directories used to store application files.)

One application may be used for planting a malicious binary in another application's trusted location. (An example is the Internet Explorer - Safari blended threat vulnerability)

The application searches for a binary in untrusted locations, possibly on remote file systems. (A typical example is a Windows application loading a dynamic link library from the current working directory after the latter has been set to a network shared folder.)

2) Blind SQL Injection

Description

Blind SQL (Structured Query Language) injection is a type of SQL Injection attack that asks the database true or false questions and determines the answer based on the applications response. This attack is often used when the web application is configured to show generic error messages, but has not mitigated the code that is vulnerable to SQL injection.

When an attacker exploits SQL injection, sometimes the web application displays error messages from the database complaining that the SQL Query's syntax is incorrect. Blind SQL injection is nearly identical to normal SQL Injection, the only difference being the way the data is retrieved from the database. When the database does not output data to the web page, an attacker is forced to steal data by asking the database a series of true or false questions. This makes exploiting the SQL Injection vulnerability more difficult, but not impossible.

3) Blind XPath Injection

Description

XPath is a type of query language that describes how to locate specific elements (including attributes, processing instructions, etc.) in an XML document. Since it is a query language, XPath is somewhat similar to Structured Query Language (SQL), however, XPath is different in that it can be used to reference almost any part of an XML document without access control restrictions. In SQL, a “user” (which is a term undefined in the XPath/XML context) may be restricted to certain databases, tables, columns, or queries. Using an XPATH Injection attack, an attacker is able to modify the XPATH query to perform an action of their choosing.

Blind XPath Injection attacks can be used to extract data from an application that embeds user supplied data in an unsafe way. When input is not properly sanitized, an attacker can supply valid XPath code that is executed. This type of attack is used in situations where the attacker has no knowledge about the structure of the XML document, or perhaps error message are suppressed, and is only able to pull once piece of information at a time by asking true/false questions(booleanized queries), much like Blind SQL Injection.

4) Session hijacking attack

Description

The Session Hijacking attack consists of the exploitation of the web session control mechanism, which is normally managed for a session token.

Because http communication uses many different TCP connections, the web server needs a method to recognize every user’s connections. The most useful method depends on a token that the Web Server sends to the client browser after a successful client authentication. A session token is normally composed of a string of variable width and it could be used in different ways, like in the URL, in the header of the http requisition as a cookie, in other parts of the header of the http request, or yet in the body of the http requisition.

The Session Hijacking attack compromises the session token by stealing or predicting a valid session token to gain unauthorized access to the Web Server.

The session token could be compromised in different ways; the most common are:

- Predictable session token;
- Session Sniffing;
- Client-side attacks (XSS, malicious JavaScript Codes, Trojans, etc);
- Man-in-the-middle attack
- Man-in-the-browser attack

5) Denial of Service

Description

The Denial of Service (DoS) attack is focused on making a resource (site, application, server) unavailable for the purpose it was designed. There are many ways to make a service unavailable for legitimate users by manipulating network packets, programming, logical, or resources handling vulnerabilities, among others. If a service receives a very large number of requests, it may cease to be available to legitimate users. In the same way, a service may stop if a programming vulnerability is exploited, or the way the service handles resources it uses.

Sometimes the attacker can inject and execute arbitrary code while performing a DoS attack in order to access critical information or execute commands on the server. Denial-of-service attacks significantly degrade the service quality experienced by legitimate users. These attacks introduce large response delays, excessive losses, and service interruptions, resulting in direct impact on availability.

6) Resource Injection

Description

This attack consists of changing resource identifiers used by an application in order to perform a malicious task. When an application defines a resource type or location based on user input, such as a file name or port number, this data can be manipulated to execute or access different resources. The resource type affected by user input indicates the content type that may be exposed. For example, an application that permits input of special characters like period, slash, and backslash is risky when used in conjunction with methods that interact with the filesystem.

The resource injection attack differs from Path Manipulation as resource injection focuses on accessing resources other than the local filesystem, while Path Manipulation focuses on accessing the local filesystem.

7) Traffic flood

Description

Traffic Flood is a type of DoS attack targeting web servers. The attack explores the way that the TCP connection is managed. The attack consists of the generation of a lot of well-crafted TCP requisitions, with the objective to stop the Web Server or cause a performance decrease.

The attack explores a characteristic of the HTTP protocol, opening many connections at the same time to attend a single requisition. This special feature of the http protocol, which consists of opening a TCP connection for every html object and closing it, could be used to make two different kinds of exploitations. The Connect attack is done during the establishment of the connection, and the Closing attack is done during the connection closing.

8) Content Spoofing

Description

Content spoofing, also referred to as content injection, “arbitrary text injection” or virtual defacement, is an attack targeting a user made possible by an injection vulnerability in a web application. When an application does not properly handle user-supplied data, an attacker can supply content to a web application, typically via a parameter value, that is reflected back to the user. This presents the user with a modified page under the context of the trusted domain. This attack is typically used as, or in conjunction with, social engineering because the attack is exploiting a code-based vulnerability and a user’s trust. As a side note, this attack is widely misunderstood as a kind of bug that brings no impact.

9) Log Injection

Description

Applications typically use log files to store a history of events or transactions for later review, statistics gathering, or debugging. Depending on the nature of the application, the task of reviewing log files may be performed manually on an as-needed basis or automated with a tool that automatically culls logs for important events or trending information. Writing invalidated user input to log files can allow an attacker to forge log entries or inject malicious content into the logs. This is called log injection.

Log injection vulnerabilities occur when:

1. Data enters an application from an untrusted source.
2. The data is written to an application or system log file.

Successful log injection attacks can cause:

1. Injection of new/bogus log events (log forging via log injection)
2. Injection of XSS attacks, hoping that the malicious log event isviewed in a vulnerable web application
3. Injection of commands that parsers (like PHP parsers) could execute

10) Setting Manipulation

Description

This attack aims to modify application settings in order to cause misleading data or advantages on the attacker's behalf. They may manipulate values in the system and manage specific user resources of the application or affect its functionalities.

An attacker can exploit several functionalities of the application using this attack technique, but it would not possible to describe all the ways of exploration, due to innumerable options that attacker may use to control the system values.

Using this attack technique, it is possible to manipulate settings by changing the application functions, such as calls to the database, blocking access to external libraries, and/or modification log files.

29/08/23

Task: Explain about any 10 web server attacks

1. **Distributed Denial of Service (DDoS)**: DDoS attacks involve overwhelming a web server with a flood of traffic from multiple sources, making it inaccessible to legitimate users. This is typically achieved using a network of compromised devices (botnets) to generate the traffic.
2. **SQL Injection**: SQL Injection is an attack where malicious SQL queries are injected into an application's input fields, exploiting vulnerabilities to manipulate or retrieve data from the underlying database. Attackers can bypass authentication, modify data, or execute unauthorized actions.
3. **Cross-Site Scripting (XSS)**: XSS attacks involve injecting malicious scripts into web applications that are then served to other users. These scripts can steal sensitive information (such as cookies), deface websites, or perform actions on behalf of the victim.

4. **Cross-Site Request Forgery (CSRF)**: CSRF attacks trick authenticated users into unknowingly performing unwanted actions on a web application by leveraging their existing session. Attackers may force users to change passwords, make unwanted purchases, or perform other actions without their consent.
5. **File Inclusion Attacks**: File Inclusion attacks exploit vulnerabilities to include malicious files from a remote server. This can lead to unauthorized access, execution of arbitrary code, or compromise of the web server's integrity.
6. **Path Traversal (Directory Traversal)**: Path Traversal attacks exploit inadequate input validation to navigate through directories and access files or directories outside the intended scope. Attackers can view sensitive files or execute unauthorized actions.
7. **Command Injection**: Command Injection attacks involve injecting malicious commands into input fields or parameters used in commands, potentially allowing an attacker to execute arbitrary commands on the server or system.
8. **Brute Force Attacks**: Brute Force attacks involve systematically trying multiple username and password combinations to gain unauthorized access to a web server. Attackers automate this process to find valid credentials.
9. **Server Misconfiguration**: Server Misconfiguration occurs when a server or its applications are improperly configured, leaving them vulnerable to attacks. Attackers can exploit misconfigured settings, permissions, or defaults to gain unauthorized access or manipulate the server.
10. **Phishing Attacks**: Phishing attacks use deceptive emails, messages, or websites that mimic legitimate entities to trick users into revealing sensitive information like login credentials, credit card details, or other personal data. These can be used to compromise web server access or perform identity theft.

30/08/23

Task: Collect info on 20 cis topics

1. **Inventory and Control of Hardware Assets:**
    - Establish and maintain an up-to-date inventory of authorized and unauthorized devices in your environment to ensure proper management and security.
2. **Inventory and Control of Software Assets:**
    - Maintain an accurate inventory of authorized and unauthorized software to effectively manage licenses, vulnerabilities, and ensure compliance.
3. **Continuous Vulnerability Management:**
    - Continuously scan, assess, and remediate vulnerabilities within the organization's systems to reduce the attack surface and improve overall security.
4. **Controlled Use of Administrative Privileges:**

- Limit and monitor access to critical systems by controlling and restricting administrative privileges to only authorized personnel.

5. **Secure Configuration for Hardware and Software:**
    - Apply and maintain secure configurations for hardware devices and software applications to mitigate known security vulnerabilities and reduce the risk of exploitation.
6. **Maintenance, Monitoring, and Analysis of Audit Logs:**
    - Regularly monitor and analyze audit logs to detect and respond to security incidents and policy violations promptly.
7. **Email and Web Browser Protections:**
    - Implement security measures to protect against email and web-based threats, such as phishing, malware, and malicious attachments, to secure user interactions with these platforms.
8. **Malware Defenses:**
    - Employ effective anti-malware solutions and practices to detect, prevent, and mitigate malware infections across all endpoints and network devices.
9. **Limitation and Control of Network Ports, Protocols, and Services:**
    - Manage network communications by restricting unnecessary ports, protocols, and services to minimize attack vectors and unauthorized access.
10. **Data Recovery Capabilities:**
    - Develop and maintain reliable and tested data backup and recovery processes to ensure critical data can be restored in the event of data loss or a cyber incident.
11. **Secure Configuration for Network Devices, such as Firewalls, Routers, and Switches:**
    - Establish secure configurations for network devices to safeguard against unauthorized access, malicious activities, and ensure proper traffic filtering and monitoring.
12. **Boundary Defense:**
    - Implement effective network security measures at organizational boundaries to detect and mitigate attacks attempting to infiltrate the network.
13. **Data Protection:**
    - Apply encryption, masking, and other mechanisms to protect sensitive data, both in transit and at rest, to maintain confidentiality and integrity.
14. **Controlled Access Based on the Need to Know:**

   - Restrict and control access to sensitive information to authorized personnel based on their specific job functions and responsibilities.

15. **Wireless Access Control:**
    - Implement security measures to control and secure wireless network access, ensuring only authorized and authenticated devices can connect.

16. **Account Monitoring and Control:**
    - Continuously monitor user accounts and activities to detect and respond to suspicious behavior or unauthorized access promptly.

17. **Implement a Security Awareness and Training Program:**
    - Educate employees about cybersecurity best practices and potential threats to promote a security-aware organizational culture.

18. **Application Software Security:**
    - Establish secure coding practices and ensure that software applications are developed, tested, and maintained with security in mind to mitigate vulnerabilities and potential exploits.

19. **Incident Response and Management:**
    - Develop and maintain an incident response plan, outlining the steps and procedures to effectively detect, respond to, and recover from security incidents.

20. **Penetration Testing and Red Team Exercises:**
    - Conduct regular penetration tests and red team exercises to identify vulnerabilities, assess the organization's security posture, and improve incident response capabilities.