cyber security

A comprehensive cheat sheet covering essential cybersecurity concepts, tools, and techniques. Includes information on threat identification, security models, ethical considerations, and common attack vectors.

Fundamentals & Principles

Core Concepts

CIA Triad:	Confidentiality, Integrity, Availability. These are the core principles of information security.
DAD Triad:	Disclosure, Alteration, and Destruction. Represents the goals of attackers against the CIA Triad.
Vulnerability:	A weakness in a system that can be exploited.
Threat:	A potential danger that can exploit a vulnerability.
Risk:	The potential for loss or damage when a threat exploits a vulnerability.
Zero Trust:	A security model based on the principle of 'never trust, always verify'.
Trust but Verify:	A security model where trust is initially granted but continuously monitored and verified.

Security Models

Bell-LaPadula Model:	Focuses on confidentiality. 'No read up, no write down'.
Biba Model:	Focuses on integrity. 'No read down, no write up'.
Clark-Wilson Model:	Focuses on integrity through well-formed transactions and separation of duty.

Principles of Privileges

Least Privilege: Granting users only the minimum necessary rights and permissions to perform their job functions.
Need to Know: Granting access to information only to individuals who require it to perform their duties.

Threats, Vulnerabilities, & Tools

Threat Identification & Intelligence

Threat Modeling: Identifying potential threats and vulnerabilities in a system.
Incident Response: The process of handling and recovering from a security incident.
Threat Intelligence: Information about potential or current attacks that can be used to prevent or mitigate them.

Threat Intelligence Classifications: Strategic, Tactical, Operational, and Technical.

The Pyramid of Pain

A model for ranking indicators of compromise (IOCs) based on their difficulty to an attacker to change. From easiest to hardest: Hashes, IP Addresses, Domain Names, Network/Host Artifacts, Tools, TTPs (Tactics, Techniques, Procedures).

Common Security Tools (CLI)

Nmap:	Network mapper for discovery and security auditing.
Metasploit:	Framework for developing and executing exploit code.
Wireshark:	Network protocol analyzer.
Aircrack-ng:	Suite of tools for assessing WiFi network security.
SQLMap:	Automatic SQL injection and database takeover tool.
Hashcat:	Password recovery tool.
Gobuster/Feroxbuster:	Directory and file discovery tools.

Common Security Tools (GUI)

Burp Suite:	Integrated platform for web application security testing.
Nessus:	Vulnerability scanner.
Autopsy:	Digital forensics platform.

Web Exploitation

Common Web Vulnerabilities

SQL Injection:	Exploiting vulnerabilities in SQL queries to gain unauthorized access to a database.
Command Injection:	Executing arbitrary commands on the server through vulnerabilities in input validation.
Cross-Site Scripting (XSS):	Injecting malicious scripts into websites to execute in the browsers of other users.
Cross-Site Request Forgery (CSRF):	Forcing a user to execute unwanted actions on a web application in which they are currently authenticated.
Insecure Direct Object Reference (IDOR):	Accessing objects by directly manipulating the object's identifier.
Server-Side Request Forgery (SSRF):	Exploiting a server-side application to make requests to unintended locations.

File Inclusion Vulnerabilities

Local File Inclusion (LFI):	Including local files on the server through a vulnerability.
Remote File Inclusion (RFI):	Including remote files on the server through a vulnerability.

Exploitation Techniques

Content Discovery: Using tools like Gobuster/Feroxbuster to find hidden files and directories.
Authentication Bypass: Techniques to circumvent authentication mechanisms.
Directory Traversal: Accessing restricted directories by manipulating file paths.

Forensics & Reverse Engineering

Forensic Analysis

File Analysis:	Examining file metadata and content to understand its purpose and origin.
PCAP Analysis:	Analyzing network traffic captures to identify malicious activity.
Steganography:	Detecting hidden messages within images, audio, or other files.
Memory Analysis:	Analyzing RAM dumps to identify running processes, injected code, and other artifacts.
Disk Imaging:	Creating a bit-by-bit copy of a storage device for forensic investigation.

Reverse Engineering

Assembly:	Low-level programming language that represents machine code.
Disassemblers & Debuggers:	Tools like IDA Pro and gdb used to analyze compiled code.
Decompilers:	Tools that attempt to convert compiled code back into a higher-level language.

Binary Exploitation

Registers: Small storage locations within the CPU used to hold data and instructions.
The Stack: A region of memory used to store local variables and function call information.
Calling Conventions: Rules that govern how functions pass arguments and return values.

Global Offset Table (GOT): A table in memory that contains the addresses of global variables.
Buffers and Buffer Overflows: Exploitable vulnerabilities that occur when data is written beyond the boundaries of a buffer.
Return Oriented Programming (ROP): A technique for executing code by chaining together small snippets of existing code.

The Heap and Exploitation: A region of memory used for dynamic allocation, often targeted for exploitation.
Format String Vulnerability: A vulnerability that allows an attacker to read from or write to arbitrary memory locations using format string functions.
Integer Overflow: A vulnerability that occurs when an integer value exceeds its maximum or minimum value.