No.

Date

Some Additional Topics

- DevSecOps Overview.
- Cloud Security and basics.
- Ethecial Hacking and Penetration Testing.

DevSecOps : Overview.

① Definition and Purpose :-

DevSecOps integrate security practices into DevOps Lifecycle, embedding cybersecurity at every stage of software development and delivery. It combines Development, security, and operations to ensure that application are built, deployed, and maintained securely without sacrificing speed or agility. The purpose of DevSecOps is to proactively address vulnerabilities, reduce risks, and deliver secure software in fast-paced, interative environments.

- **Integration of security into DevOps Life-cycle :-**

→ Integration of security into DevSecOps Life-cycle involves embedding security practices throughout the software development and operations processes.

-Key Points :-

① Security is incorporated from the earliest stages (shift-left approach)

No.

Date

(2) Automated security testing (SAST, DAST) runs in CI/CD pipelines.

(3) IaC (Infrastructure as Code) is scanned for vulnerabilities

(4) Continuous monitoring detects runtime threats

(5) Collaboration ~~monitoring det~~ among development, security, and operations teams ensures shared responsibility.

(6) Security policies and compliance checks are enforced automatically.

(7) Tools integrate seamlessly to provide real-time feedback and prevent insecure code deployment.

Key Components :-

1. Secure Code Practices
2. Automated Security Testing
3. Infrastructure as Code (IaC)
4. CI/CD Pipelines with security gates.

No.

Date

Benefits :-

(Helps in organization Prioritizing Cyber Security).

① Early Detection of Vulnerabilities :- By integrating security early (Shift-left), DevSecOps identifies vulnerabilities during coding or testing, reducing the likelihood of exploitable flaws reaching production. This pro-active approach minimizes security incidents.

② Faster, Secure Software Delivery :-
Automation and collaboration streamline security processes, enabling rapid development cycles without compromising safety. Teams can release software faster while maintaining high security standards.

③ Reduced Risk & Cost :- Fixing vulnerabilities early is significantly cheaper than addressing them post-deployment. DevSecOps reduce the risk of breaches, data loss etc. saving organization from costly remediation.

PTO.

No.

Date

• Challenges :- ① Cultural Changes

② Tool Integration Complexity ③ Skills & Training Requirements

• Tools & Technologies

Eg of Tools ① Jenkins : Open Source CI/CD tool

② Gitlab → A CI/CD platform with built in security scanning for code

③ SonarQube : A SAST tool that analyze code for vulnerabilities and code quality issues

④ Aqua Security :- Scans Containers and cloud workload for misconfiguration & vulnerabilities.

⑤ HashiCorp :- Manages secrets and credentials securely, preventing unauthorized access.

• Use/Case :- (Eg :-) Financial Services (capital one).

No.

Date

→ Capital One, a major bank, adopted DevSecOps to secure its cloud-based application. By integrating SAST- & DAST into CI/CD pipelines and using AWS-native security tools, they reduced vulnerabilities by 50% and accelerated secure deployments, ensuring compliance with strict financial regulation

Eg→ E-Commerce (Etsy) :→ Etsy implemented DevSecOps to protect its online Platform. Using automated security testing and continuous monitoring, they embedded security into their agile development process. This enabled rapid features releases while maintaing robust defenses against threats like data breaches

Eg→ Healthcare (Philips) :- Philips adopted DevSecOps to secure medical software and devices. By incorporating secure coding practices & IaC. Security scans, they ensured compliance with HIPAA regulations, reducing vulnerabilities in critical health care system.

Ethical Hacking + Penetration Testing (Overview).

No.
Date

⊙ Intro: Definition of Ethical Hacking :- It involves authorized attempts to ~~ethical~~ identify and exploit vulnerabilities in system, network or application to improve their security.

→ ⊙ Strengthen security, ⊙ Identifying vulnerabilities and fixing issues. ⊙ Prevent financial loss.

→ Types of Hackers already done (✓).

Ethical - Hacking (Process)

Hacking follows a structured, systematic process to ~~sta~~ simulate real-world cyber-attacks while adhering to legal and ethical boundaries. This process identifies vulnerabilities, tests defense and provides actionable insights to strengthen an organization's security posture.

The 5 key phases → ① Reconnaissance ② Scanning ③ Enumeration, ④ Gaining access, ⑤ Maintaining access & Covering Tracks.

No.

Date

① Reconnaissance :- Reconnaissance, often called 'recon', is the inital phase, where ethical hackers gathers information about the target system, network, or organization. The phase mimics how malicious hackers plan attacks by collecting data to identify potential entry points. Reconnaissance is divided into → ① Passive & ② Active methods.

① Passive ⇒ Uses publicly available sources without direct interaction, such as analyzing a company's website, social media, or WHOIS records to gather details about domains, IP addresses, or employee roles. Eg → an Hacker might use LinkedIn to identify key personnel for social engineering attacks or review for exposed code.

② Active ⇒ Involves direct interaction, such as querying DNS servers or performing limited network probes, but ethical hackers must stay within the agreed scope to avoid legal issues. Tools like Maltego or Shodan help aggregate data, while techniques like Hacking Google (using advanced search operators) uncover sensitive information, such as exposed login pages or configuration files. This phase is critical for building a detailed attack surface map without alerting target.

Ethical Hacking + Penetration Testing (Overview)

No. Date

⊙ Intro: Definition of Ethical Hacking :- It involves authorized attempts to ~~Ethical~~ identify and exploit vulnerabilities in system, network or application to improve their security.

→ ⊙ Strengthen security, ⊙ Identifying vulnerabilities and fixing issues. ⊙ Prevent financial loss.

→ Types of Hackers already done (✓).

▲ Ethical - Hacking (Process)

Hacking follows a structured, systematic process to ~~sta~~ simulate real-world cyber-attacks while adhering to legal and ethical boundaries. This process identifies vulnerabilities, tests defense and provides actionable insights to strengthen an organization's security posture.

The 5 key phases → ① Reconnaissance ② Scanning ③ Enumeration, ④ Gaining access, ⑤ Maintaining access & Covering Tracks.

No.

Date

① Reconnaissance :- Reconnaissance, often called 'recon', is the inital phase, where ethical hackers gathers information about the target system, network, or organization. The phase mimics how malicious hackers plan attacks by collecting data to identify potential entry points. Reconnaissance is divided into → ① Passive & ② Active methods.

① Passive ⇒ Uses publicly available sources without direct interaction, such as analyzing a company's website, social media, or WHOIS records to gather details about domains, IP addresses, or employee roles. Eg → an Hacker might use LinkedIn to identify key personnel for social engineering attacks or review for exposed code.

② Active ⇒ Involves direct interaction, such as querying DNS servers or performing limited network probes, but ethical hackers must stay within the agreed scope to avoid legal issues. Tools like Maltego or Shodan help aggregate data, while techniques like Hacking Google (using advanced search operators) uncover sensitive information, such as exposed login pages or configuration files. This phase is critical for building a detailed attack surface map without alerting target.

No.

Date

(2) Scanning and Enumeration :- Scanning & Enumeration involve actively probing the target to identify vulnerabilities and gather detailed system information. Scanning uses tools like Nmap to discover open ports, running services, and operating systems, revealing potential weaknesses, such as unpatched software, or misconfigured firewalls. For instance, an ethical hacker might find an outdated Apache server vulnerable to known exploits. Enumeration goes deeper, extracting specific details like user accounts, network shares, or application version. Tools like EnumAlinux or NetBIOS help enumerate Windows system, while Nessus scans for known vulnerabilities. This phase requires careful calibration to avoid triggering intrusion detection sys.
- Ethical hackers analyze scan results to prioritize vulnerabilities based on exploitability, ensuring the subsequent phase focus on high-impact weaknesses. For eg → discovering an open port 22 (SSH), with weak credentials could lead to a targeted attack simulation.

(3) Gaining Access :- In this phase, ethical hackers attempt to exploit identified vulnerabilities to gain unauthorized access simulating real attacker techniques. Methods include exploiting software flaws (eg> SQL injection in web application), cracking weak password with tools like hydra,

No.

Date

or leveraging misconfigurations, such as overly permissive access control. For eg → an ethical hacker might use Metasploit to exploit a known vulnerability in a web server, gaining a foothold. Social engineering, like phishing emails, may also be used to trick users into revealing credentials. The goal is to test how far an attacker could penetrate and what data or system could be compromised.

④ Maintaining Access :- Once access is gained, ethical hackers test whether an attacker could maintain a persistent presence. This involves creating backdoors, installing covert tools, or modifying user accounts to simulate prolonged control. For instance, an ethical hacker might create a hidden user account or deploy a reverse shell to maintain remote access. Tools like Netcat or Meterpreter facilitate this phase. The objective is to evaluate the organization's ability to detect and respond to persistent threats.

⑤ Covering Tracks :- The final phase tests an organization's detection

Date

Capabilities by simulating how attackers hide their activities. Ethical hackers clear logs, delete temporary files, or modify timestamps to avoid detection by security tools like SIEM systems. For eg → they might use scripts to wipe access logs on a compromised server. This phase highlights gaps in monitoring and logging, helping organization improve IR.

Ethical Hackers meticulously document their actions, ensuring transparency, and provide recommendations to enhance detection mechanisms., such as implementing robust log retention policies or anomaly detection system.

Tools used

Reconnaissance :- ① Shodan, ② Maltego ③ Recon-ng, ④ theharvester

Eg. • Using Maltego to map a company's domain to its IP addresses and identify hosting providers.

① Leveraging google hacking (eg → Site:target.com filetype:Pdf). to find exposed sensitive docs.

② Use shodan to find misconfigured web servers with open Port.

No.

Date

② Scanning and Enumeration :=>

- Tools :=> ① Nmap.
  ② Nessus.
  ③ Enum4linux.
  ④ OpenVAS.

eg> Running nmap -sv target.com to identify an outdated server on port 80.

② Using Nessus to detect a vulnerable version of SMB prone to Eternal Blue exploit

③ Enumerating user acc on windows server with Enum4linux to find weakly protected accounts.

③ Gaining Access :- ① Metasploit
② Hydra
③ SQLMap, ④ BurpSuite.

eg> Exploiting a web application's SQL injection flow with SQLmap to access a database.

) Using Hydra to get weak SSH credentials.

④ Maintaining Access ① Netcat, ② MeterPeretes
③ Mimikatz, ④ Cron jobs.

→ Setting netcat listener to maintain a reverse shell on a compromised server.

→ Scheduling a malicious script via Cron to ensure periodic access to a linux host.

⑤ Covering Tracks :- Tools → ① CCleaner
② Logrotate
③ Event log clear
④ Timestomp.

No.

Date

eg. ① Using a script to clear Apache access logs (/var/log/access.log) on a linux server.

② Running wevtutil cl system to clear Windows system event logs

③ Modifying file timestamps with Timestomp to hide evidence of file access.

→ Tools used & their Function :-

1. ① Nmap :- Network Mapper is versatile tool for network discovery and security auditing. It scans for open ports, services, and operating system, mapping a network's attack surface.

2. ② Metasploit :- A penetration testing platform for developing, testing and executing exploits. It includes modules for vulnerabilities like MS17-010 (eternal-Blue).

3. Wireshark :- A packet analyzer for capturing and inspecting network traffic. It helps identify anomalies, such as unencrypted credentials

No.

Date

④ Burp Suite :- A web application testing tool for intercepting and manipulating HTTP/HTTPS requests. It's used to find vulnerabilities like Cross-Site scripting (XSS) or SQL injection.

⑤ Aircrack-ng :- A suite for auditing wireless networks, capable of cracking WEP/WPA keys.

⑥ John the Ripper :- A password-cracking tool for testing weak credentials. It supports brute force and dictionary attacks

⑦ SQL map :- Automates SQL injection attacks to test database vulnerabilities. Eg :- Extracting user data from a vulnerable web application database.

⑧ Cain & Abel :- A Windows-based tool for password recovery and network sniffing.

⑨ Kali Linux :- A specialized Linux distribution pre loaded with hacking tools like Nmap, Metasploit, Burp suite.

⑩ Hashcat :- A high-performance password cracking tool optimized for GPU usage.

No.

Date

(11) Hydra :- A password-cracking tool for brute-forcing services like SSH, FTP

(12) Nessus : A vulnerability scanner for identifying weaknesses like outdated software or misconfigurations.

- Techniques →

⊙ Social Engineering
⊙ MiTM

→ ⊙ Exploits :- Leverage software or system vulnerabilities to gain access.

→ ⊙ Vulnerability Scanning :- Uses automated tools to identify weaknesses - like unpatched software or open ports.

→ ⊙ Password Attacks :- Involve cracking or guessing credentials. Techniques include brute-forcing, dictionary attacks or rainbow table attacks

→ ⊙ Web-Application-Testing : Targets web apps for vulnerabilities like XSS

Date

→ SQL injection, or insecure APIs.

→ ⊙ Wireless Attacks :- Toggets Wi-Fi networks to crack encryption or spoof access points

→ ⊙ Privilege Escalation :- Gains higher-level access after initial entry

→ ⊙ Network Sniffing :- Captures network traffic to extract sensitive data.

⊙ Penetration Testing Basics :-

Penetration testing is controlled, authorized simulation of cyber attacks aimed at evaluating the security of system, networks, or application. By mimicking real-world attacker techniques, Pen testing identifies vulnerabilities, assesses their exploitability, and validates the effectiveness of security control. It helps organization prioritize remediation, strengthen defenses and comply with regulatory requirements. Pen testing differs from ethical hacking in its structured, goal-oriented approach, often focusing on specific

No.

Date

System as objective defined by a scope agreement. This section explores the definition, objectives, types and methodologies of Pen-testing.

Objectives :-

→ ① Identifying Vulnerabilities :- Discovering flaws like unpatched software, misconfigurations, or weak passwords that could be exploited.

→ ② Assessing Impact :- Evaluating the potential damage of a succesful attack, such as data breaches, or system down time.

③ Validating Control :→ Testing the effectiveness of firewalls, Intrusion detection systems (IDS), or encryption.

→ ④ Ensuring Compliance :- Meeting standards like PCI-DSS, HIPAA or GDPR which often mandate regular Pen tests.

No.

Date

① Types of Pentration Testing :-

→ Penetration tests vary based on the tester's knowledge of the target system, each simulating different attacker perspectives.

① Black-Box Testing :- Testers have no prior knowledge of the system, mimicking external attackers. This approach tests real-world scenarios, such as an outsider targeting a public-facing website.
(eg→ A black box test might involve scanning a company's web application for SQL injection flaws without its backend structure. While realistic, it may miss internal vulnerabilities due to limited access.

② White Box Testing :- Testers have full knowledge of system, including source code, architecture, or credentials. This enables thorough testing internal components, ideal for identifying deep-seated flaws. For instance, a white box test might analyze a web application's code to find insecure API calls. It's resource-intensive but highly comprehensive

Date

- **Grey-Box Testing :-** Testers have partial knowledge, such as user credentials or network diagram, simulating insider threats or attackers with limited access. For eg, a grey box test might use a low-privilege account to attempt privilege escalation. This balances realism and depth., making it efficient for many scenarios. Each type aligns with specific goals, such as testing external defenses (black-box) or internal config- (white box), and organization choose based on risk profiles and resources.

Methodologies :-

→ Standardized methodologies ensure consistent, repeatable pen tests.

- **OWASP :-** (Open Web Application Security Project)
  :→ Focuses on web application testing, addressing risks like SQL injection and XSS.

- **PTES :-** (Penetration Testing Execution Standard)
  :→ A comprehensive framework covering all pen-testing phases, from planning to reporting, applicable to

networks, application, and system.

- Other methodologies, are like ① NIST SP 800-115 or OSSTMM, provide additional structure, emphasizing compliance or operational security.

→ These frameworks ensure tests are systematic, ethical and aligned with organizational goals, reducing the risk of oversight or harm.

(Just Overview) ....

- Basis list of types of Cyber-attack.

1) Network	2) Application	3) Social Engineer attacks
① DoS	① SQL injection	① Phishing
② DDoS	② XSS ③ CSRF	② Whaling
③ MitM	④ Command injection	③ Tailgating
④ Packet Sniffing	⑤ Directory Traversal	④ Pretexting
⑤ IP Spoofing	⑥ File Inclusion	⑤ Click Jacking
⑥ ARP	⑦ Zero Day	⑥ Spear Phishing
⑦ DNS	⑧ Business logic Attack	⑦ Baiting

PTO.

No.

Date

(4) Authentication & Password Attacks (5)

(1) Brute force, (2) Dictionary Attack
(3) Credential stuffing, (4) Keylogging
(5) Password Spraying

(5) Malware attacks (6)

(1) Virus, (2) Trojan, (3) Worm, (4) Ransomware
(5) Spyware, (6) Rootkit.

(6) Cloud-Specific Attacks

(1) Cloud Malware injection.
(2) Account Hijacking.
(3) Insecure APIs.
(4) Misconfiguration Exploits.
(5) Data Breach.

(7) Physical & Insider Attacks

(1) USB Drop
(2) Hardware Tampering
(3) Dumpster Diving
(4) Shoulder Surfing
(5) Malicious Insider.

(Basic ends here)!! ☺