ಮಂಗಳೂರು ವಿಶ್ವವಿದ್ಯಾನಿಲಯ

MANGALORE UNIVERSITY

(Accredited by NAAC Grade)

ಕ್ರಮಾಂಕ/No. : MU/ACC/CR.24/2025-26/A2

ಕುಲಸಚಿವರಕಛೇರಿ
ಮಂಗಳಗಂಗೋತ್ರಿ – 574 199
Office of the Registrar
Mangalagangothri – 574 199
ದಿನಾಂಕ/Date:31.07.2025

NOTIFICATION

Sub: Revised syllabus of M.Sc. in Cyber Security Programme.

Ref: Academic Council approval vide agenda No.:ಎಸಿಸಿ:ಶೈ.ಮ.ಸಾ.ಸ.1:1 (2025-26) dtd 18.07.2025.

*****

The revised syllabus of M.Sc. in Cyber Security Programme which has been approved by the Academic Council at its meeting held on 18.07.2025 is hereby notified for implementation with effect from the academic year 2025-26 and onwards.

Copy of the Syllabus shall be downloaded from the University Website (www.mangaloreuniversity.ac.in)

REGISTRAR

To,

1. The Registrar (Evaluation), Mangalore University.
2. The Chairman, UG and PG Combined Board of Studies in Electronics, Dept. of Electronics, Mangalore University.
3. The Chairman, Dept. of Electronics, Mangalore University.
4. The Asst. Registrar (ACC), O/o the Registrar, Mangalore University.
5. The Director, DUIMS, Mangalore University – with a request to publish in the website.
6. Guard File.

M.Sc. Cybersecurity Syllabus

(July 2025, B-series)

Vision Statement

To sculpt cyber guardians of tomorrow—ethical, insightful, and battle-ready—who defend digital realms with wisdom, wield algorithms with purpose, and rise as sentinels of security in an evolving, interconnected world.

We envision a future where cybersecurity is not just a profession, but a **responsibility**—where every graduate is equipped not only with tools and techniques, but also with the **clarity of thought**, **strength of ethics**, and **grit of defenders** who safeguard trust in the digital age.

Mission Statement

Our mission is to:

1. **Deliver hands-on, real-world cybersecurity education** rooted in both foundational theory and cutting-edge practice.
2. **Foster algorithmic thinking, secure development habits, and digital forensics expertise** across diverse platforms and threat landscapes.
3. **Integrate governance, law, risk, and ethics** to shape cybersecurity professionals who lead with integrity and insight.
4. **Build a curriculum that adapts to emerging technologies** like AI, blockchain, and cyber threat intelligence—ensuring lifelong relevance.
5. **Cultivate critical thinkers and ethical hackers**, capable of defending infrastructure, decoding threats, and inspiring digital resilience in industry and society.

What the Revision Embodies

This curriculum revision is not a patch—it's a **re-architecture,** where:

- **Labs simulate reality**, not rehearsed exercises
- **Threat intelligence is lived**, not memorized
- **Laws are explored with reflection**, not rote
- **Students become creators of security**, not just consumers

It reflects your belief that the future of cybersecurity must be **practical**, **ethical**, and **intellectually rich**—a journey from blackboard to battlefield, from byte to justice.

Programme Specific Objectives:

1. **To build foundational understanding** of cybersecurity principles, ethical practices, and threat landscapes in an interconnected digital world.
2. **To cultivate hands-on technical mastery** in areas such as network defense, system hardening, ethical hacking, forensics, malware analysis, and cloud security.
3. **To develop strong analytical and algorithmic thinking**, preparing students to understand, model, and combat emerging cyber threats.
4. **To enable students to design and evaluate security policies, protocols, and architectures** for complex systems across industries.
5. **To bridge the knowledge-action gap** through experiential learning via labs, projects, simulations, and internships.
6. **To instill cross-disciplinary awareness** by including aspects of law, governance, compliance, and risk management relevant to cybersecurity.
7. **To prepare students for dynamic cybersecurity careers** in industry, research, public sectors, or entrepreneurial ventures with ethical integrity and global awareness.

Programme Outcomes

By the end of the program, students will be able to:

1. **Understand and apply cybersecurity concepts and frameworks** (e.g., CIA triad, threat models, risk analysis, NIST standards).
2. **Design, develop, and test secure systems and applications** using modern tools, programming languages, and secure coding practices.
3. **Analyze and evaluate algorithms** for their efficiency and role in cryptographic and cybersecurity mechanisms.
4. **Investigate cyber incidents** using digital forensics, malware analysis, and threat intelligence techniques.

5. **Identify and mitigate vulnerabilities** in networks, systems, and applications through ethical hacking and red teaming.
6. **Implement and manage security solutions in cloud and IoT environments**, considering real-world constraints and performance trade-offs.
7. **Interpret legal, ethical, and compliance requirements** and integrate them into cybersecurity planning and operations.
8. **Communicate findings, strategies, and risks effectively** to both technical and non-technical stakeholders.
9. **Contribute independently or collaboratively** in multidisciplinary teams, managing projects and adapting to evolving technologies.
10. **Pursue lifelong learning** and advanced research in cybersecurity domains.

SEMESTER I			
SL. NO	**COURSE NAME**		**CREDITS**
HARD CORE			
1	CSCH 401B	CYBERSECURITY PRINCIPLES & PRACTICES	4
2	CSCH 402B	PYTHON PROGRAMMING FOR CYBERSECURITY	3
3	CSCH 403B	COMPUTER NETWORKS	3
SOFT CORE			
4	CSCS 404B	ALGORITHMIC THINKING FOR SECURITY	3
5	CSCS 405B	LINUX ADMINISTRATION & SHELL SCRIPTING	3
PRACTICALS			
9	CSCP 409B	PYTHON AND LINUX PROGRAMMING LAB	3
10	CSCP 410B	COMPUTER SYSTEM AND COMPUTER NETWORKS LAB	3
		TOTAL	**22**

SEMESTER II			
SL. NO	**COURSE NAME**		**CREDITS**
HARD CORE			
1	CSCH 451B	CYBER LAW, GOVERNANCE, AND RISK MANAGEMENT	3
2	CSCH 452B	ETHICAL HACKING	3
3	CSCH 453B	NETWORK SECURITY	3
SOFT CORE			
4	CSCS 454B	CRYPTOGRAPHY : ALGORITHMS AND APPLICATIONS	3
5	CSCS 455B	CLOUD COMPUTING & SECURITY	3
PRACTICALS			
8	CSCP 459B	NETWORK SECURITY LAB	3
9	CSCP 460B	ETHICAL HACKING LAB	3

		OPEN CHOICE	
11	CSCO 462B	CYBER AWARENESS & DIGITAL HYGIENE	3
		TOTAL	**24**

SEMESTER III			
SL. NO	**COURSE NAME**		**CREDITS**
HARD CORE			
1	CSCH 501B	CYBER THREAT INTELLIGENCE	3
2	CSCH 502B	MALWARE ANALYSIS	3
3	CSCH 503B	DATA MANAGEMENT & SECURITY	3
SOFT CORE			
4	CSCS 504B	DIGITAL FORENSICS	3
5	CSCS 505B	WEB DEVELOPMENT & SECURE CODING	3
PRACTICALS			
8	CSCP 509B	CYBER THREAT INTELLIGENCE LAB	3
9	CSCP 510B	MALWARE ANALYSIS LAB	3
		OPEN CHOICE	
11	CSCO 512B	CRIMES, CYBERCRIME, DIGITAL EVIDENCE AND CYBER LAW	3
		TOTAL	**24**

Scheme of Examination for M. Sc. in Cyber Security

Semester I

Course Code	**Title of the Course**	**Credits**	**Hours per week**	**Duration of the Exam**	**Marks**		
					IA	**Exam**	**Total**
Hard Core							
CSCH 401B	Cybersecurity Principles & Practices	04	04	03 hours	30	70	100
CSCH 402B	Python Programming for	03	03	03 hours	30	70	100

	Cybersecurity						
CSCH 403B	Computer Networks	03	03	03 hours	30	70	100
Soft Core							
CSCS 404B	Algorithmic Thinking for Security	03	03	03 hours	30	70	100
CSCS 405B	Linux Administration & Shell Scripting	03	03	03 hours	30	70	100
Practicals							
CSCP 409B	Python and Linux Programming Lab	03	03	03 hours	30	70	100
CSCP 410B	Computer System and Computer Networks Lab	03	03	03 hours	30	70	100
Total					**210**	**490**	**700**

Semester II

Course Code	Title of the Course	Credits	Hours per week	Duration of the Exam	Marks		
					IA	Exam	Total
Hard Core							
CSCH 451B	Cyber Law, Governance and Risk Management	03	03	03 hours	30	70	100
CSCH 452B	Ethical Hacking	03	03	03 hours	30	70	100
CSCH 453B	Network Security	03	03	03 hours	30	70	100
Soft Core							
CSCS 454B	Cryptography : Algorithms and Applications	03	03	03 hours	30	70	100
CSCS 455B	Cloud Computing & Security	03	03	03 hours	30	70	100
Practicals							
CSCP 459B	Network Security Lab	03	03	03 hours	30	70	100
CSCP 460B	Ethical Hacking Lab	03	03	03 hours	30	70	100
Open Choice							
CSCO 462B	Cyber Awareness & Digital Hygiene	03	03	3 hours	30	70	100
Total					**240**	**560**	**800**

Semester III

Course Code	Title of the Course	Credits	Hours per week	Duration of the Exam	Marks		
					IA	Exam	Total
Hard Core							
CSCH 501B	Cyber Threat Intelligence	03	03	3 hours	30	70	100
CSCH 502B	Malware Analysis	03	03	3 hours	30	70	100
CSCH 503B	Data Management & Security	03	03	3 hours	30	70	100
Soft Core							
CSCS 504B	Digital Forensics	03	03	3 hours	30	70	100

CSCS 505B	Web Development & Secure Coding	03	03	3 hours	30	70	100
Practicals							
CSCP 509B	Cyber Threat Intelligence Lab	03	06	03 hours	30	70	100
CSCP 510B	Malware Analysis Lab	03	06	03 hours	30	70	100
Open Choice							
CSCO 512B	Crimes, Cybercrime, Digital Evidence and Cyber Law	03	03	03 hours	30	70	100
Total					**240**	**560**	**800**

Semester IV

Course Code	**Title of the Course**	**Credits**	**Hours per week**	**Duration of the Exam**	**Marks**		
					IA	**Exam**	**Total**
Hard Core							
CSCH 551B	Application Security	03	03	03 hours	30	70	100
CSCH 552B	Incident Response & Security Operations	03	03	03 hours	30	70	100
CSCH 553B	AI in Cybersecurity	04	04	03 hours	30	70	100
Soft Core							
CSCS 554B	Blockchain, Security & Use Cases	03	03	3 hours	30	70	100
CSCS 555B	IT Audit	03	03	3 hours	30	70	100
CSCP 560B	Internship+Dissertation (In-Campus)	06	06	12	60	140	200
				Total	210	490	700

Marks Distribution Semester Wise

Semester	**Credits**	**Marks**
I	22	700
II	24	800
III	24	800
IV	22	700
Total	**92**	**3000=00**

SEMESTER IV			
SL. NO	**COURSE NAME**		**CREDITS**
HARD CORE			
1	CSCH 551B	APPLICATION SECURITY	3
2	CSCH 552B	INCIDENT RESPONSE & SECURITY OPERATIONS	3
3	CSCH 553B	AI IN CYBERSECURITY	4
SOFT CORE			
4	CSCS 554B	BLOCKCHAIN, SECURITY & USE CASES	3
5	CSCS 555B	IT AUDIT	3
6	CSCP 560B	INTERNSHIP + DISSERTATION (IN-CAMPUS)	6
TOTAL			22

SEMESTER – I	
CSCH 401B	**CYBERSECURITY PRINCIPLES & PRACTICES**

Course Objectives

By the end of this course, the learner will:

1. Understand foundational principles of cybersecurity including confidentiality, integrity, availability, and the extended models used in digital trust.
2. Recognize and classify various cyber threats, vulnerabilities, and risk factors across modern technological environments.
3. Develop knowledge of key security mechanisms including access control, cryptographic essentials, system hardening practices, and architectural security.
4. Apply best practices in designing secure environments using principles such as defense in depth, least privilege, and secure configuration.
5. Critically evaluate the ethical, legal, and social implications of cybersecurity decisions, especially in the face of evolving threats and emerging technologies.

Course Outcomes

Upon successful completion of the course, students will be able to:

1. Explain core cybersecurity principles and threat models, and apply them to

real-world digital systems.

2. Identify and analyze common attack vectors and develop suitable security responses using controls and frameworks.
3. Implement basic cybersecurity policies and recommend configurations for system hardening and protection.
4. Demonstrate understanding of ethical, legal, and risk-related dimensions of cybersecurity through case studies and policy analysis.
5. Engage in informed decision-making about emerging cybersecurity challenges using foundational knowledge, standards, and ethical reasoning.

UNIT I: Foundations of Cybersecurity *(16 Hours)*

"Before we wield the sword, we must understand why it is forged."

Security Triad (CIA) and Extensions (Parkerian Hexad), Types of Threats and Attack Vectors, Security Goals and Trust Models, Threat Landscape: Personal, Enterprise, Nation-State, Introduction to Vulnerability, Risk & Impact, Asset Identification and Classification, Security Policies & Standards: ISO/IEC 27001, NIST, CIS, Case Studies: Classic Breaches (e.g., Equifax, SolarWinds)

Outcome: Students articulate what cybersecurity protects, from whom, and why

UNIT II: Security Controls, Mechanisms, and Models *(16 Hours)*

"A castle isn't secure because it's tall—but because it's intelligently guarded."

Access Control Models: MAC, DAC, RBAC, ABAC, Authentication and Authorization Mechanisms, Firewalls, IDS/IPS, VPNs, and Endpoint Security, Physical Security vs Logical Security, Cryptographic Basics: Hashes, Keys, Encryption Types, Security Design Principles: Least Privilege, Defense in Depth, Fail-Safe Defaults, Security Frameworks: OWASP Top 10, NIST Cybersecurity Framework, Hands-on Simulation: Classify and Apply Controls to Sample Scenarios

Outcome: Students will be able to categorize, evaluate, and apply control mechanisms.

UNIT III: Risk, Ethics, and Future Trends *(16 Hours)*

"Security without ethics is like a shield with no wielder."

Risk Management Lifecycle: Assessment, Mitigation, Monitoring, Business Continuity and Disaster Recovery Basics, Ethics in Cybersecurity: Whistleblowing, Surveillance, Consent, Legal Overview: Data Protection, Privacy Acts (GDPR, Indian IT Act), Security Operations and Incident Response Primer, Emerging Challenges: AI, Quantum Threats, Deepfakes, Security Metrics & Reporting, Case Debate: Ethical Dilemmas in Cyber Defense

Outcome: Students will reason critically, weigh risk, and recognize ethical stakes.

Primary Textbooks

1. **William Stallings** – ***Effective Cybersecurity: A Guide to Using Best Practices and Standards***
   Pearson Education
   ➤ Offers comprehensive coverage of NIST, ISO, and foundational security principles with real-world application. Ideal for Unit I & II.
2. **Chuck Easttom** – ***Computer Security Fundamentals*** **(4th Edition)**
   Pearson / Pearson IT Certification
   ➤ Beginner-friendly yet technical. Strong on CIA triad, access control, network defense, and legal frameworks.
3. **Michael E. Whitman & Herbert J. Mattord** – ***Principles of Information Security*** **(6th Edition)**
   Cengage Learning
   ➤ Rich with diagrams, case studies, and end-of-chapter exercises. Well-suited for academic instruction.

Reference Books

1. **Anderson, Ross** – ***Security Engineering: A Guide to Building Dependable Distributed Systems***
   Wiley
   ➤ Deep dives into design principles, cryptography, protocols. Best used as a reference for high-performing students.
2. **Eric Cole** – ***Network Security Bible***
   Wiley
   ➤ A good reference for control mechanisms and layered defense.
3. **Bruce Schneier** – ***Secrets and Lies: Digital Security in a Networked World***
   Wiley
   ➤ Philosophical and practical insights into real-world cybersecurity beyond pure tech—great for UNIT III discussions.

Online Resources and Toolkits

1. **NIST Cybersecurity Framework**
   ➤ https://www.nist.gov/cyberframework
   The authoritative standard for security architecture, risk, and resilience.
2. **OWASP Top Ten & Security Knowledge Framework**
   ➤ https://owasp.org
   Hands-on security flaws, secure design principles, real-world applications.
3. **Cybrary Courses & Labs (Free & Paid)**
   ➤ https://www.cybrary.it
   Bite-sized courses for visual learners, especially helpful for hands-on concepts.

4. **MITRE ATT&CK® Framework**
   ➤ https://attack.mitre.org
   Mapping of attacker behaviors and real-world case analysis—perfect for Unit III.

Supplementary Media

- **YouTube Channels:**
  - *Computerphile* (explains core cybersecurity concepts in plain English)
  - *The Cyber Mentor* (ethical hacking, real-world attacks)
  - *HackerSploit* (free cybersecurity tutorials)
- **Podcasts:**
  - *CyberWire Daily Briefing*
  - *Smashing Security*
  - *Darknet Diaries* (brilliant storytelling on real-world hacks and ethics)

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Explain core cybersecurity principles and threat models, and apply them to real-world digital systems.	✓	✓								
CO2 – Identify and analyze common attack vectors and develop suitable security responses using controls and frameworks.	✓	✓	✓		✓					
CO3 – Implement basic cybersecurity policies and recommend configurations for system hardening and protection.	✓	✓			✓	✓	✓			
CO4 – Demonstrate understanding of ethical, legal, and risk-related dimensions of cybersecurity through case studies and policy analysis.							✓	✓		
CO5 – Engage in informed decision-making about emerging cybersecurity challenges using foundational knowledge, standards, and ethical reasoning.	✓		✓	✓		✓	✓	✓	✓	✓

SEMESTER – I

CSCH 402B	Python Programming for Cybersecurity

Course Objectives

By the end of this course, the learner will:

1. Master Python fundamentals required for cybersecurity scripting and automation.
2. Use Python to analyze, extract, and process data from files, logs, and networks.
3. Build simple tools and scripts that aid in real-world cybersecurity operations.
4. Apply programming to reconnaissance, hashing, encryption, and packet analysis.
5. Demonstrate ethical responsibility and legal awareness in building and deploying security scripts.

Course Outcomes

Upon successful completion of this course, students will be able to:

1. Write efficient Python code for system and file-level automation in cybersecurity.
2. Perform data parsing, hashing, and log analysis using Python libraries.
3. Design scripts for scanning, monitoring, and information gathering.
4. Leverage open APIs and threat intelligence feeds within their code.
5. Demonstrate an understanding of ethical boundaries and responsible coding in cybersecurity contexts.

UNIT I: Python Foundations for Security Applications *(16 Hours)*

> *"First, we learn to wield the serpent before commanding it to guard the realm."*

Python Syntax and Semantics, Data Types and Structures (lists, sets, dictionaries, tuples), Control Flow (if-else, loops), Functions and Scope, Modules and Packages, File Handling and Input/Output, Exception Handling, Working with External Libraries and pip, Pythonic Style (PEP8, best practices)

Outcome: Students will comfortably write well-structured, reusable, and modular Python code.

UNIT II: Python for Cybersecurity Automation and Analysis *(16 Hours)*

"Now, let the serpent roam—with purpose, through networks and files."

Working with OS and Subprocess modules for automation, Reading and parsing logs, Regular Expressions for pattern matching, Hashing using hashlib and hmac, File integrity checking, Working with JSON, XML, and Config files, Network programming with socket, Automating reconnaissance tasks (port scanning, banner grabbing)

Outcome: Students will build automation scripts for basic cybersecurity tasks including scanning and monitoring.

UNIT III: Security Tooling, Scripting, and Ethical Use (16 Hours)

"Let your tools be ethical, your scripts accountable, and your output clear."

Web scraping and threat intel extraction (using requests, BeautifulSoup), Working with APIs (VirusTotal, Shodan, AbuseIPDB), Packet sniffing basics using Scapy, Working with password lists and brute force logic, Basic encryption/decryption using PyCryptodome, Log parsing and alert generation, Reporting with matplotlib, pandas, Legal and ethical considerations in security scripting

Outcome: Students will develop their own Python-based cybersecurity toolkit with an understanding of responsible use.

Textbooks & Resources

Primary Textbooks

- **Burt Harris & David Clinton** – ***Python for Cybersecurity: Using Python for Cyber Offense and Defense***
  Wiley
  ➤ Excellent for offensive + defensive scripting with examples.
- **TJ O'Connor** – ***Violent Python: A Cookbook for Hackers, Forensic Analysts, Penetration Testers and Security Engineers***
  Elsevier
  ➤ A hacker's cookbook—legendary and practical.

Supplementary References

- Al Sweigart – *Automate the Boring Stuff with Python*
  (freely available at: automatetheboringstuff.com)
- Justin Seitz – *Black Hat Python*
- GitHub Repositories:
  ➤ https://github.com/topics/python-cybersecurity

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Write efficient Python code for system and file-level	✓	✓	✓							✓

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
automation in cybersecurity.										
CO2 – Perform data parsing, hashing, and log analysis using Python libraries.	✓	✓	✓	✓						
CO3 – Design scripts for scanning, monitoring, and information gathering.	✓	✓		✓	✓					
CO4 – Leverage open APIs and threat intelligence feeds within their code.	✓	✓		✓	✓	✓		✓	✓	
CO5 – Demonstrate an understanding of ethical boundaries and responsible coding in cybersecurity contexts.							✓	✓		✓

SEMESTER – I

CSCH 403B	**Computer Networks**

Course Objectives

By the end of this course, learners will:

1. Understand core concepts of data communication, networking layers, and architecture.
2. Analyze network devices and their roles in switching, routing, and connectivity.
3. Explore network addressing, protocols, and topologies with practical examples.
4. Gain awareness of network services and their vulnerabilities.
5. Prepare for advanced courses in network security, ethical hacking, and cloud systems.

Course Outcomes

Upon completion, students will be able to:

1. Describe OSI and TCP/IP models and map protocols to each layer.
2. Differentiate between types of networks and their components.
3. Configure IP addressing and subnetting for basic networks.
4. Analyze and monitor basic network traffic using tools.

5. Identify common network vulnerabilities and describe defensive measures.

UNIT I: Fundamentals of Networking and Internet Architecture (12 Hours)

> *"Where the physical meets the digital, the wires begin to whisper."*

Introduction to Networking Concepts, Types of Networks: LAN, MAN, WAN, Internet vs Intranet vs Extranet, Network Topologies and Protocols, OSI Model: Layers and Functions, TCP/IP Stack and Comparison with OSI, Addressing: IP, MAC, Subnetting Basics, DNS and DHCP Mechanisms

Outcome: Students will grasp the layered approach to networking and the flow of data through protocols.

UNIT II: Devices, Switching, and Routing Mechanisms (12 Hours)

> *"The path of a packet is not straight—it is wisely chosen by the guardians of the network."*

Network Devices: Hub, Switch, Router, Bridge, Gateway, NAT and PAT Concepts, Switching Techniques: Circuit, Packet, Message, IP Routing Basics and Protocols (RIP, OSPF, BGP), ARP and ICMP, IPv4 vs IPv6, Wireless Networking Basics: Wi-Fi Protocols, SSIDs, Authentication

Outcome: Students will understand how data is routed, switched, and moved through various devices securely and efficiently.

UNIT III: Network Services, Security, and Performance (12 Hours)

> *"No network is complete without guardians—protocols of trust, filters of protection, and gates of policy."*

Client-Server and Peer-to-Peer Models, Ports and Sockets, FTP, HTTP, HTTPS, SMTP, Telnet, SSH Protocols, Firewalls and Proxy Servers, Network Performance Metrics: Latency, Throughput, Jitter, Packet Loss, Introduction to VPN, IDS/IPS Basics, Packet Sniffing, DoS Attacks and Defense

Outcome: Students will gain working knowledge of services, performance concerns, and introductory security mechanisms.

Textbooks and Resources

Primary Textbooks

- **Andrew S. Tanenbaum & David J. Wetherall – *Computer Networks* (5th Edition)**
  Pearson
  ➤ A classic, covering all layers with theory and illustrations.
- **Behrouz A. Forouzan – *Data Communications and Networking* (5th Edition)**
  McGraw-Hill

➤ Beginner-friendly and visually rich, great for non-engineering students.

Supplementary Resources

- **William Stallings – *Data and Computer Communications***
- Cisco Networking Academy (Free resources & simulations)
  ➤ https://www.netacad.com
- Packet Tracer Simulation Tool
- Wireshark for packet analysis (used later in labs)

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Describe OSI and TCP/IP models and map protocols to each layer.	√									
CO2 – Differentiate between types of networks and their components.	√					√				
CO3 – Configure IP addressing and subnetting for basic networks.	√	√								
CO4 – Analyze and monitor basic network traffic using tools.	√	√		√	√					
CO5 – Identify common network vulnerabilities and describe defensive measures.	√	√		√	√	√	√			

SEMESTER – I

CSCS 404B	ALGORITHMIC THINKING FOR SECURITY

Course Objectives

By the end of this course, learners will:

1. Develop foundational algorithmic thinking and problem-solving skills in the context of cybersecurity.
2. Understand and implement core algorithmic paradigms such as recursion, divide & conquer, greedy algorithms, dynamic programming, and backtracking.
3. Analyze algorithm complexity and understand time-space trade-offs in security scenarios.
4. Apply algorithmic design to common cybersecurity use cases: cryptography, intrusion detection, access control, forensics, and malware analysis.

5. Explore computational hardness, brute-force limitations, and cryptographic reductions as part of security reasoning.

Course Outcomes (COs)

Upon successful completion of the course, students will be able to:

1. **Explain** algorithmic paradigms and **identify suitable strategies** for security problems.
2. **Analyze and compare** time and space complexity for various algorithms.
3. **Apply algorithmic thinking** to develop security tools and techniques.
4. **Implement algorithms** for detection, encryption, access control, and secure search.
5. **Evaluate the feasibility** of solving security problems within computational limits.

Unit I: Foundations of Algorithmic Reasoning

"The enemy's logic is your map to defense."

Problem-solving strategies, Pseudocode, Time complexity, Space complexity, Recursion, Backtracking, Sorting algorithms, Searching algorithms, Hashing techniques, Brute-force methods, Efficiency in password cracking

Outcome: Students will gain confidence in breaking down problems and analyzing efficiency.

Unit II: Algorithmic Paradigms in Cybersecurity

"To secure the system, one must first secure the logic."

Divide and conquer, Greedy algorithms, Dynamic programming, Graph algorithms, Shortest path, Network traversal, Trust models, Dijkstra's algorithm, Breadth-first search (BFS), Depth-first search (DFS), Threat propagation models.

Outcome: Students will implement algorithmic models that reflect real-world digital defense strategies.

Unit III: Security Algorithms and Computational Limits

"Not everything computable is computable in time."

Cryptographic algorithms, Modular arithmetic, RSA, ECC basics, Key exchange protocols, Hash functions, Pattern matching algorithms, KMP, Rabin-Karp, Bloom filters, Merkle trees, NP-completeness, Cryptographic reductions, Secure search techniques

Outcome: Students will recognize the inherent limits of computation in security and design accordingly.

Textbooks & References

Core Textbooks

- Cormen et al. – *Introduction to Algorithms* (CLRS)
- Dasgupta, Papadimitriou – *Algorithms*
- Eric Demaine – *Algorithmic Thinking* (for intuitive design)

Supplementary

- Vazirani – *Foundations of Cryptography*
- Neil C. Rowe – *Introduction to Cybersecurity Algorithms*
- Khan Academy – Algorithms Series

Tools & Languages

- Language: Python (preferred), with occasional C++ or pseudo-code
- Libraries: hashlib, heapq, networkx, numpy

Platforms: LeetCode, HackerRank, CyberChef, Codeforces

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – **Explain** algorithmic paradigms and **identify suitable strategies** for security problems.	√									
CO2 – **Analyze and compare** time and space complexity for various algorithms.	√					√				
CO3 – **Apply algorithmic thinking** to develop security tools and techniques.	√	√								
CO4 – **Implement algorithms** for detection, encryption, access control, and secure search.	√	√		√	√					
CO5 – **Evaluate the feasibility** of solving security problems within computational limits.	√	√		√	√	√	√			

SEMESTER – I

CSCS 405B	LINUX ADMINISTRATION & SHELL SCRIPTING

Course Objectives

This course aims to:

1. Introduce the Linux operating system for system administration tasks.
2. Develop command-line and shell scripting skills to automate tasks.
3. Enable real-world administration through scripting, logs, and scheduled jobs.
4. Prepare students to manage services and monitor performance securely.
5. Provide foundational skills for advanced security, forensics, and DevOps.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Administer users, processes, and file systems on Linux systems.
2. Write functional shell scripts for automation of routine tasks.
3. Schedule jobs, rotate logs, and maintain system uptime.
4. Apply disk and network tools to monitor and optimize system performance.
5. Automate and secure basic system activities with shell scripting.

UNIT I: Linux Fundamentals and System Administration (12 Hours)

"The shell is your wand, the system your spellbook."

Linux Architecture and Distributions (Ubuntu, Fedora, Kali), Boot Process and Init Systems (systemd), Linux File System Structure (/etc, /var, /home, /dev), User and Group Management (adduser, usermod, passwd), File and Directory Permissions (chmod, chown, umask), Package Management (APT, YUM, RPM), Process Management (ps, top, kill, nice), Introduction to Logging (/var/log, journalctl)

Outcome: Students will navigate, configure, and manage a Linux environment confidently.

UNIT II: Shell Scripting Basics and Automation (12 Hours)

"Repetition is for the mortal. Automation is for the wise."

Shell Types (sh, bash, zsh), Writing and Executing Scripts, Variables, I/O with echo and read, Control Flow: if, case, Loops: for, while, until, Functions and Modular Scripts, Reading from Files, Command Substitution ($(), backticks), Debugging with set -x, Use of Environment Variables

Outcome: Students will write scripts to automate user tasks and streamline system operations.

UNIT III: System Automation, Scheduling, and Security (12 Hours)

"A good admin sleeps peacefully—because the system is scripted wisely."

Disk Management Tools (df, du, mount, lsblk), Backup Scripting (tar, rsync), Cron Jobs and at Scheduling, Service Control (systemctl, service), Network Commands

(ifconfig, ip, netstat, ss), Log Rotation and Monitoring (logrotate), Scripting Security Tasks: User Watchers, Log Cleaners, Basic Firewall Management (iptables, ufw)

Outcome: Students will design scripts to automate auditing, backups, and system security routines.

Textbooks and References

Primary Textbooks

1. **Mark G. Sobell** – ***A Practical Guide to Linux Commands, Editors, and Shell Programming***
   Prentice Hall
   Comprehensive and accessible; ideal for theory + scripting.
2. **Richard Blum** – ***Linux Command Line and Shell Scripting Bible***
   Wiley
   Excellent for scripting-heavy components; great lab reference.

Reference Books

- **Evi Nemeth et al.** – ***UNIX and Linux System Administration Handbook***
  Pearson
  Industry-standard reference; sysadmin heavy.
- **Jason Cannon** – ***Linux for Beginners***
  Especially useful for non-CS students new to terminal work.

Supplementary Online Resources

- https://linuxjourney.com – Interactive Linux learning paths
- https://explainshell.com – Visual explanation of any shell command
- https://tldr.sh – Community-based simplified Linux command cheat sheets
- YouTube: *NetworkChuck*, *The Cyber Mentor*, *DorianDotSlash*

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Administer users, processes, and file systems on Linux systems.	√	√				√				
CO2 – Write functional shell scripts for automation of routine tasks.	√	√							√	√
CO3 – Schedule jobs, rotate logs, and maintain system uptime.	√	√		√		√				
CO4 – Apply disk and network tools to monitor and optimize system performance.	√	√		√	√	√				
CO5 – Automate and secure	√	√			√	√	√			

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
basic system activities with shell scripting.										

SEMESTER – I	
CSCP 409B	**PYTHON AND LINUX PROGRAMMING LAB**

Course Objectives

This lab aims to:

1. Reinforce foundational skills in Python and Linux scripting via hands-on tasks.
2. Enable automation of cybersecurity-related workflows and administrative functions.
3. Develop practical capabilities in writing Python and Bash scripts to interact with files, networks, and logs.
4. Integrate system-level knowledge (Linux) with logic-based programming (Python).
5. Foster habit-forming scripting practices vital for red team, blue team, and SecOps roles.

Course Outcomes (COs)

Upon successful completion of the lab, students will be able to:

1. Develop and debug Python scripts to automate basic cybersecurity tasks.
2. Write shell scripts to manage users, monitor systems, and process logs.
3. Integrate Python and Bash to build hybrid automation utilities.
4. Perform common sysadmin operations using custom scripts in real-world Linux environments.
5. Document and present code with clarity and purpose for security tasks.

Part A: Python Programming Lab (16 Hours)

1. **Basic File Analysis Script** – Read, parse, and summarize contents of a system log file.
2. **Hash Checker Utility** – Use hashlib to detect tampering in files.
3. **Simple Port Scanner** – Using socket module to check for open ports on a given IP.
4. **Password Strength Checker** – Validate length, entropy, and known weak

patterns.

5. **Threat Feed Extractor** – Scrape a security blog for recent CVEs using requests and BeautifulSoup.
6. **Log Aggregator Tool** – Read multiple logs and output summary of failed login attempts.
7. **Brute Force Simulation (with limits)** – Try password combos on a dummy password (no real attack).
8. **Visualization of Login Patterns** – Use matplotlib to plot login times or IPs.

Part B: Linux Shell Scripting Lab (16 Hours)

1. **User Creation Script** – Add user accounts with custom permissions.
2. **Disk Usage Monitor** – Warn when /home exceeds 80% using df, awk.
3. **Automated Backup Script** – Use tar, rsync to back up /etc/ to a USB or remote directory.
4. **Scheduled Script with Cron** – Rotate logs or run a disk monitor every 6 hours.
5. **Scripted Firewall Setup** – Configure basic iptables rules to block or allow services.
6. **Service Health Checker** – Ping service ports (22, 80, etc.) and report status.
7. **System Audit Log Extractor** – Pull and format last 50 logins and failed logins.
8. **Interactive Menu Script** – Build a menu to run system checks or initiate backups.

Mini Integration Project Ideas

Combine Python + Linux scripting

- **Hybrid Alert System**: Use Python to scan logs and call a Bash script that sends email alerts.
- **Security Toolkit**: Package multiple scripts (port scanner, log parser, firewall rules) into a simple CLI.
- **Scheduled Threat Visualizer**: A Python script scheduled by cron to plot daily login activity.

Books & Resources

Primary Lab References

1. **Mark G. Sobell** – ***A Practical Guide to Linux Commands, Editors, and Shell Programming***
   Pearson / Prentice Hall

2. **Richard Blum** – ***Linux Command Line and Shell Scripting Bible***
   Wiley
3. **Burt Harris & David Clinton** – ***Python for Cybersecurity: Using Python for Cyber Offense and Defense***
   Wiley

Supplementary Materials

- **Online Platforms**:
  - https://www.hackerrank.com/domains/tutorials/10-days-of-javascript (for Python scripting practice)
  - https://linuxjourney.com
  - https://tldr.sh – command cheatsheets
  - Explainshell.com – great for Bash command visualization
- **Kali Linux or Ubuntu VMs** for safe practice
- GitHub Templates for log parsers and shell dashboards

Course Outcomes (COs)	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Develop and debug Python scripts to automate basic cybersecurity tasks.	√	√								√
CO2 – Write shell scripts to manage users, monitor systems, and process logs.	√	√		√	√	√				
CO3 – Integrate Python and Bash to build hybrid automation utilities.	√	√		√	√	√			√	√
CO4 – Perform common sysadmin operations using custom scripts in real-world Linux environments.	√	√		√	√	√				
CO5 – Document and present code with clarity and purpose for security tasks.	√						√	√	√	√

SEMESTER – I	
CSCP 410B	**COMPUTER SYSTEM AND COMPUTER NETWORKS LAB**

Course Objectives

This lab course is designed to:

1. Provide hands-on understanding of computer hardware components and basic system-level diagnostics.
2. Train students in the configuration and troubleshooting of LANs and networking devices.
3. Introduce packet behavior through simulation and analysis tools.
4. Develop working knowledge of IP addressing, port scanning, and protocol-level diagnostics.
5. Bridge concepts from system hardware to real-world networking configurations.

Course Outcomes (COs)

Upon successful completion of this lab, students will be able to:

1. Identify and assemble key components of a computer system.
2. Configure and test small-scale network environments including addressing and cabling.
3. Use diagnostic and monitoring tools to analyze network traffic.
4. Perform basic network administration using CLI and GUI tools.
5. Simulate protocol behaviors and packet flow using visual and textual tools.

Part A: Computer Systems & Diagnostics

1. **Assemble a Basic Desktop Computer** – Identify motherboard, CPU, RAM, HDD, power supply
2. **BIOS/UEFI Settings Exploration** – Boot order, virtualization, fan settings
3. **OS Installation (Windows/Linux Dual Boot)** – Partitioning and file system handling
4. **Hardware Troubleshooting** – Use tools like lshw, dmidecode, Device Manager
5. **Peripheral Management** – Connect and troubleshoot printers, USBs, webcams
6. **System Monitoring** – Use top, taskmgr, htop, iostat
7. **Disk Health & Partitioning** – Use fdisk, gparted, chkdsk or fsck
8. **System Performance Tuning** – Basic service management and boot-up optimization

Part B: Computer Networking

1. **Basic LAN Setup** – Connect multiple systems via switch/router; assign IPs
2. **IP Addressing and Subnetting Exercises** – Calculate and configure

3. **Ping and Traceroute** – Understand path and hops between devices
4. **Port Scanning with Nmap / Zenmap** – Identify open ports and services
5. **Network Traffic Analysis with Wireshark** – Capture HTTP, DNS, and TCP handshakes
6. **DNS and DHCP Simulation** – Using tools like Packet Tracer / GNS3
7. **Network Configuration in Linux/Windows** – CLI and GUI setup
8. **Simulate DoS Scenario (Legally Safe)** – Using hping3 or slowloris on test VMs

Mini Project Ideas

- **Build a LAN + Internet Sharing with Firewall Rules**
- **Simulate Packet Routing Between Subnets**
- **Network Inventory Tool** – Scan and log all connected devices with their IPs/MACs
- **Latency Monitor** – Python + Ping automation script with graph output

Textbooks and Lab Manuals

Primary Lab Books

1. **Behrouz A. Forouzan – *Data Communications and Networking* (5th Edition)**
   McGraw Hill – Core network behavior, simulation-ready explanations
2. **Mark Minasi – *Mastering Windows Network Administration***
   Sybex – Friendly, practical walkthrough of configuration tasks
3. **Michael Jang – *Linux Networking Cookbook***
   O'Reilly – CLI-based tasks, great for Linux network diagnostics

Reference Books

- **Andrew S. Tanenbaum – *Computer Networks***
- **Seymour Lipschutz – *Data Structures in C/C++*** *(if hardware-level memory understanding is needed)*

Supplementary Tools & Platforms

- **Cisco Packet Tracer** (for simulated networking exercises)
- **Wireshark** (packet sniffing)
- **GNS3** (advanced network simulation)
- **Nmap/Zenmap** (port scanning)
- **IPCalc / Subnet Calculator**

- **Open-source OS Images**: Ubuntu Server/Desktop, Kali, Fedora Workstation

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Identify and assemble key components of a computer system.	✓	✓								
CO2 – Configure and test small-scale network environments including addressing and cabling.	✓	✓	✓		✓					
CO3 – Use diagnostic and monitoring tools to analyze network traffic.	✓	✓	✓	✓					✓	
CO4 – Perform basic network administration using CLI and GUI tools.	✓	✓	✓	✓					✓	
CO5 – Simulate protocol behaviors and packet flow using visual and textual tools.	✓	✓	✓	✓					✓	✓

SEMESTER – II

CSCH 451B	CYBER LAW, GOVERNANCE, AND RISK MANAGEMENT

Course Objectives

This course aims to:

1. Introduce the legal foundations underpinning cybersecurity enforcement and regulation.
2. Develop an understanding of governance models in digital institutions.
3. Teach practical frameworks for risk assessment, response, and strategic security planning.
4. Build awareness of national and international cyber policies.
5. Prepare students to bridge technical knowledge with compliance and legal responsibility.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Interpret and apply key cyber laws, especially within Indian and global legal frameworks.
2. Identify various types of cybercrime and understand legal procedures of

redressal.

3. Evaluate cybersecurity governance frameworks and their role in organizational defense.
4. Perform risk assessments and propose mitigation strategies using industry best practices.
5. Integrate legal, ethical, and governance perspectives into security planning and policy making.

UNIT I: Cyber Law and Legal Frameworks (12 Hours)

Need for Cyber Law and Digital Governance, IT Act 2000 and 2008 Amendments (India), Cybercrime Classifications: Financial, Personal, Sovereign, Jurisdiction in Cyberspace, International Law & Treaties, Case Studies on Legal Action in Cybercrime, Legal Liability of ISPs, Intermediaries, and Platforms, Digital Evidence: Admissibility and Chain of Custody, Role of CERT-In, Interpol, and Cyber Tribunal

Outcome: Students gain foundational legal awareness and jurisdictional clarity.

UNIT II: Cyber Governance, Policy & Compliance (12 Hours)

Principles of Cybersecurity Governance, Information Security Policy and Frameworks (ISO 27001, NIST, COBIT), Roles of CISO, DPO, SOC, and Governance Committees, National Cybersecurity Policy (India, select global models), Data Governance and Sovereignty (India's Data Protection Laws, GDPR), Vendor & Supply Chain Governance, Internal Audit and External Certification, Government vs Corporate Governance Models

Outcome: Students understand how governance models shape security maturity and accountability.

UNIT III: Risk Management & Strategic Security Planning (12 Hours)

Risk Management Principles and Lifecycle, Threat Intelligence & Vulnerability Mapping, Risk Analysis: Qualitative vs Quantitative, Risk Appetite, Exposure, and Mitigation Strategies, Risk Registers and Risk Heat Maps, Business Continuity and Disaster Recovery, Cyber Insurance and Legal Risk Transfer, Incident Response Planning and Post-Incident Auditing

Outcome: Students will gain tools to assess, prioritize, and act on risks effectively.

Books and Resources

Primary Textbooks

1. **Pavan Duggal** – ***Cyber Law: The Indian Perspective***
   Universal Law Publishing
   ➤ Authoritative and updated with Indian IT Act coverage.

2. **Raef Meeuwisse** – ***Cybersecurity for Beginners & Governance Guide***
   ➤ Blends governance and risk in accessible terms.
3. **K.K. Bajaj and Debashis Chakrabarty** – ***Cyber Law & Information Technology***
   ➤ Detailed legal provisions and case law.

Supplementary References

- **Thomas R. Peltier** – ***Information Security Policies, Procedures, and Standards***
- **Paul Rohmeyer** – ***Cyber Risk Management*** (Springer)
- **ISACA Manuals (COBIT)**, **NIST 800-30**, and **ISO/IEC 27005 Risk Guidelines**

Online Resources

- CERT-In Guidelines and Circulars
- https://meity.gov.in (Ministry of Electronics & IT, India)
- https://nvlpubs.nist.gov – for all NIST security and risk standards
- https://gdpr.eu – EU's General Data Protection Regulation

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Interpret and apply key cyber laws...	√				√	√	√	√		√
CO2 – Identify cybercrimes and legal redressal procedures	√				√	√	√	√		√
CO3 – Evaluate governance frameworks...	√	√			√	√	√	√	√	√
CO4 – Perform risk assessments and mitigation strategies	√	√	√	√	√	√	√		√	√
CO5 – Integrate legal, ethical, and governance perspectives	√	√	√	√	√	√	√	√	√	√

SEMESTER – II

CSCH 452B	**ETHICAL HACKING**

Course Objectives

This course aims to:

1. Introduce ethical hacking as a structured and professional security testing process.
2. Explore real-world attack techniques and the security principles that defend

against them.

3. Provide insight into hacker mindsets, tools, and phases of compromise.
4. Develop critical thinking for vulnerability identification, penetration strategies, and defense reporting.
5. Emphasize ethical, legal, and responsible disclosure practices.

Course Outcomes (COs)

By the end of this course, students will be able to:

1. Describe ethical hacking phases and contrast them with malicious hacking techniques.
2. Conduct reconnaissance and scanning using open-source intelligence (OSINT) and tools.
3. Identify and exploit common system and web vulnerabilities.
4. Simulate post-exploitation behaviors and recommend effective countermeasures.
5. Draft ethical hacking reports, document vulnerabilities, and propose remediations responsibly.

UNIT I: Ethical Hacking Foundations and Reconnaissance (12 Hours)

Ethical Hacking Overview and Importance, Types of Hackers and Legal Considerations, Phases of Ethical Hacking: Reconnaissance, Scanning, Gaining Access, Maintaining Access, Covering Tracks, OSINT Tools and Techniques, Footprinting using WHOIS, Netcraft, Google Dorking, DNS Enumeration, Social Engineering Fundamentals, Passive vs Active Reconnaissance

Outcome: Understand hacker mindset, motivation, and early-phase tools.

UNIT II: Scanning, Exploitation and Vulnerability Analysis (12 Hours)

Network Scanning using Nmap and Zenmap, Port Discovery and Banner Grabbing, Vulnerability Scanning Tools: OpenVAS, Nessus, Metasploit Framework Basics, Password Cracking with John the Ripper and Hydra, Exploiting System Vulnerabilities, Web Exploits: SQL Injection, XSS, CSRF, Command Injection, Buffer Overflow (conceptual), Privilege Escalation Techniques

Outcome: Develop offensive skills for detecting and exploiting common vulnerabilities.

UNIT III: Post-Exploitation, Countermeasures and Reporting (12 Hours)

Maintaining Access: Backdoors, Keyloggers, Remote Access Tools, Log Manipulation and Covering Tracks, Packet Sniffing with Wireshark, Security

Countermeasures and Patch Management, Risk Rating with CVSS, Vulnerability Documentation and Exploit Proof-of-Concepts, Report Writing and Responsible Disclosure, Case Studies of Real-world Ethical Hacks

Outcome: Understand how to responsibly conduct, contain, and report ethical assessments.

Textbooks and Reference Material

Primary Textbooks

1. **Sean-Philip Oriyano** – ***Ethical Hacking and Countermeasures: EC-Council Series***
   Cengage Learning
2. **Georgia Weidman** – ***Penetration Testing: A Hands-On Introduction to Hacking***
   No Starch Press

Reference Books & Supplementary

- **David Kennedy et al.** – ***Metasploit: The Penetration Tester's Guide***
- **OWASP Testing Guide** (latest version online)
- **NIST SP 800-115 – Technical Guide to Information Security Testing**
- **Kali Linux Revealed – Offensive Security** (Free online book)
- Hacker101 Web Platform, HackTheBox, TryHackMe Labs

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Describe ethical hacking phases...	√				√	√	√			√
CO2 – Conduct reconnaissance and scanning using OSINT	√	√	√	√	√				√	
CO3 – Identify and exploit vulnerabilities	√	√	√	√	√				√	
CO4 – Simulate post-exploitation behaviors and countermeasures	√	√	√	√	√	√			√	√
CO5 – Draft reports, document vulnerabilities, propose remediations	√			√	√	√	√	√	√	√

SEMESTER – II	
CSCH 453B	**NETWORK SECURITY**

Course Objectives

This course aims to:

1. Introduce core principles and models of network security.
2. Teach students how to identify, assess, and mitigate common network threats.
3. Familiarize learners with network defense tools and configuration strategies.
4. Explore modern trends in securing wired, wireless, and cloud-based communications.
5. Build readiness for advanced certification tracks in networking and security.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Explain key concepts of network security, including confidentiality, integrity, and availability.
2. Analyze common network attacks and develop basic defense mechanisms.
3. Configure firewalls, access control lists, and VPNs for secured communication.
4. Apply authentication and encryption protocols to networked systems.
5. Evaluate real-world security architectures and incident response strategies.

UNIT I: Network Security Concepts and Threat Landscape (12 Hours)

Introduction to Network Security Principles, Confidentiality, Integrity, and Availability (CIA), Threats: DoS, DDoS, Spoofing, MITM, Session Hijacking, Malware Propagation Over Networks, TCP/IP Vulnerabilities, Security Policy and Posture, Intrusion Detection and Prevention Concepts, Security in LAN, WAN, and Wireless Environments, Wireless Threats: Rogue APs, Eavesdropping, Evil Twin

Outcome: Learners gain conceptual clarity and threat awareness across various network scopes.

UNIT II: Secure Protocols and Device Hardening (12 Hours)

Cryptographic Protocols in Network Security: SSL/TLS, IPSec, SSH, HTTPS, AAA Framework: RADIUS and TACACS+, Device Security: Routers, Switches, Wireless Controllers, Network Address Translation (NAT), Port Security, VLAN Security, IP Filtering and ACLs, DHCP Snooping and ARP Inspection, Network Segmentation

Outcome: Students understand protocol-level defenses and device-specific hardening.

UNIT III: Network Defense Tools and Incident Handling (12 Hours)

Firewall Architecture and Configuration (Stateless vs Stateful), IDS and IPS Tools (Snort, Suricata), VPN Concepts and Configuration (IPSec, SSL VPN), SIEM Overview and Log Analysis Basics, Honeypots and Honeynets, Risk Management in Network Design, Incident Response Phases, Case Studies in Breach Containment, Securing Cloud-based and Virtual Networks

Outcome: Learners will deploy practical defense systems and understand incident response cycles

Textbooks & Resources

Primary Textbooks

1. **William Stallings** – ***Network Security Essentials: Applications and Standards***
   Pearson
2. **Behrouz A. Forouzan** – ***Data Communications and Networking*** **(Selected Security Chapters)**
   McGraw Hill

Reference Books

- **Chris Sanders** – ***Practical Packet Analysis*** **(Wireshark-focused)**
- **Mike Chapple** – ***CompTIA Security+ Guide to Network Security Fundamentals***
- **NIST Special Publications**: SP 800-41 (Firewalls), SP 800-94 (IDS/IPS)
- **Cisco Network Security Essentials Manuals** (For ACL, VPN configs)

Online Tools & Supplementary Materials

- Cisco Packet Tracer
- GNS3 for simulation
- Wireshark Lab Manual
- https://nvlpubs.nist.gov – for compliance-based readings
- https://owasp.org – for network-level best practices

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Explain key concepts of network security	√				√					
CO2 – Analyze common network attacks and develop defense	√	√	√	√	√	√			√	
CO3 – Configure firewalls, ACLs, and VPNs	√	√	√	√	√				√	√

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO4 – Apply authentication and encryption protocols	√	√	√	√	√				√	
CO5 – Evaluate real-world security architectures and response	√	√	√	√	√	√		√	√	√

SEMESTER – II	
CSCS 454B	**CRYPTOGRAPHY : ALGORITHMS AND APPLICATIONS**

Course Objectives

This course aims to:

1. Introduce fundamental concepts in cryptography and its historical evolution.
2. Explain symmetric and asymmetric cryptographic algorithms with mathematical reasoning.
3. Teach applications of cryptography in securing data, communications, and authentication.
4. Explore cryptographic protocols used in modern cybersecurity systems.
5. Build foundational readiness for applied cryptanalysis and secure protocol design.

Course Outcomes (COs)

Upon successful completion of the course, students will be able to:

1. Explain core principles and historical development of cryptography.
2. Analyze and implement symmetric and asymmetric cryptographic algorithms.
3. Apply cryptographic techniques for confidentiality, integrity, and authentication.
4. Evaluate digital signatures, hash functions, and key exchange protocols.
5. Interpret cryptographic applications in secure communication, certificates, and blockchain.

UNIT I: Foundations of Cryptography and Classical Ciphers (12 Hours)

Historical Background and Evolution of Cryptography, Cryptographic Goals: Confidentiality, Integrity, Authentication, Non-repudiation, Caesar Cipher, Vigenère

Cipher, Monoalphabetic and Polyalphabetic Ciphers, One-Time Pad, Modular Arithmetic, Number Theory Essentials, Euler's Theorem, Fermat's Little Theorem, Introduction to Computational Complexity in Cryptography

Outcome: Students will understand the philosophy, purpose, and building blocks of ciphers.

UNIT II: Symmetric and Asymmetric Algorithms (12 Hours)

Block and Stream Ciphers, DES and AES Structures, Key Scheduling, Feistel Networks, Modes of Operation (ECB, CBC, CTR), Public Key Cryptography Concepts, RSA Algorithm, Key Generation and Proofs, Diffie-Hellman Key Exchange, Elliptic Curve Cryptography (Introductory Level), Security of RSA and ECC

Outcome: Learners will apply both symmetric and asymmetric algorithms in practical settings.

UNIT III: Cryptographic Applications and Protocols (12 Hours)

Message Authentication Codes (MAC), Hash Functions (MD5, SHA-1, SHA-256), Digital Signatures and PKI, Certificate Authorities and SSL/TLS Handshake, PGP and Secure Email, Cryptographic Protocols for Authentication (Kerberos, OAuth), Blockchain Fundamentals and Cryptographic Proofs, Quantum Cryptography (Overview Only)

Outcome: Students will see cryptography in action across applications, from email to blockchain.

Textbooks and References

Primary Textbooks

1. **William Stallings** – ***Cryptography and Network Security: Principles and Practice***
   Pearson
   Comprehensive and industry-standard for core theory + examples.
2. **Behrouz A. Forouzan** – ***Cryptography and Network Security***
   McGraw-Hill
   Very friendly for Indian learners; includes mathematical appendices.

Reference Books

- **Douglas Stinson** – ***Cryptography: Theory and Practice***
- **Bruce Schneier** – ***Applied Cryptography***
- **Christof Paar and Jan Pelzl** – ***Understanding Cryptography***
- **Katz & Lindell** – ***Introduction to Modern Cryptography*** (advanced reference)

Supplementary Materials

- Online Number Theory Tools: https://www.desmos.com/scientific, https://www.geogebra.org/numbertheory
- Python Libraries: pycryptodome, hashlib, ecdsa
- Interactive Playground: https://cryptohack.org, https://cryptopals.com

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Explain core principles and history of cryptography	√				√					
CO2 – Analyze and implement cryptographic algorithms	√	√	√	√	√				√	
CO3 – Apply techniques for confidentiality, integrity, and authentication	√	√	√	√	√				√	
CO4 – Evaluate digital signatures, hash functions, key exchange	√	√	√	√	√				√	√
CO5 – Interpret applications in secure comm., certificates, blockchain	√	√	√	√	√	√		√	√	√

SEMESTER – II

CSCS 455B	**CLOUD COMPUTING & SECURITY**

Course Objectives

This course aims to:

1. Introduce the fundamental concepts, models, and architecture of cloud computing.
2. Explain the security challenges and strategies in cloud environments.
3. Familiarize learners with service models (IaaS, PaaS, SaaS) and deployment types.
4. Teach access control, encryption, and compliance in distributed systems.
5. Develop awareness of virtualization, multi-tenancy, and zero-trust cloud security models

Course Outcomes (COs)

Upon successful completion of the course, students will be able to:

1. Describe cloud computing models, architecture, and deployment strategies.

2. Identify key security threats and controls in cloud environments.
3. Analyze authentication, access management, and virtualization security.
4. Apply compliance, auditing, and governance policies to cloud-based services.
5. Evaluate best practices for securing IaaS, PaaS, and SaaS models in real-world use cases.

UNIT I: Cloud Fundamentals and Architecture (12 Hours)

Cloud Computing Characteristics and Evolution, Service Models: IaaS, PaaS, SaaS, Deployment Models: Public, Private, Hybrid, Community, Cloud Reference Architecture (NIST), Virtualization Concepts: Hypervisor, Containers, Multi-tenancy and Resource Isolation, Cloud Storage Models, Key Enabling Technologies: Grid, Utility, SOA, Edge Computing (Intro)

Outcome: Understand architecture and foundational cloud building blocks.

UNIT II: Cloud Security Concepts and Threats (12 Hours)

Cloud Security Issues: Data Breach, Insecure APIs, Account Hijacking, Multi-tenancy Threats, Identity and Access Management in Cloud, Authentication and Authorization in Cloud Services, Virtual Machine Security and Isolation, Network Security in Cloud, Zero-Trust Security Model, Shared Responsibility Model

Outcome: Students will identify threats and explore architectural safeguards.

UNIT III: Compliance, Governance, and Best Practices (12 Hours)

Cloud Security Standards: ISO/IEC 27017, NIST 800-144, CSA Controls Matrix, Cloud Risk and Compliance: PCI-DSS, HIPAA, GDPR, Cloud Audit and Assessment Tools, Logging and Forensics in Cloud, SLA and Contractual Security, Cloud Security Policies and Incident Response Planning, Case Studies: AWS, Azure, Google Cloud

Outcome: Students will be able to assess governance frameworks and regulatory practices in cloud setups.

Textbooks & Resources

Primary Textbooks

1. **Tim Mather, Subra Kumaraswamy** – ***Cloud Security and Privacy***
   O'Reilly Media
2. **Rajkumar Buyya** – ***Cloud Computing: Principles and Paradigms***
   Wiley

Reference Books

- **Ronald L. Krutz** – ***Cloud Security: A Comprehensive Guide to Secure Cloud Computing***

- **Thomas Erl – *Cloud Computing: Concepts, Technology & Architecture***
- **Shroff – *Virtualization and Cloud Security* (for Indian market alignment)**

Supplementary Materials

- **NIST SP 800-144**, **NIST 800-145** (cloud security and definitions)
- **Cloud Security Alliance (CSA)**: https://cloudsecurityalliance.org
- **AWS Free Tier**, **Azure Student Portal**, **Google Cloud Free Labs**
- **Linux Foundation's LFS452 (Online)** – Essentials of Cloud Security

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Describe cloud models and architecture	√				√					
CO2 – Identify security threats and controls in cloud	√	√	√	√	√				√	
CO3 – Analyze authentication, access, and virtualization security	√	√	√	√	√	√			√	
CO4 – Apply compliance, auditing, and governance policies	√	√		√	√	√	√	√	√	√
CO5 – Evaluate best practices for securing IaaS, PaaS, SaaS	√	√	√	√	√	√	√	√	√	√

SEMESTER – II

CSCP 459B	NETWORK SECURITY LAB

Course Objectives

This lab aims to:

1. Provide practical experience in configuring, securing, and testing networked environments.
2. Enable students to use tools for traffic analysis, firewall design, and protocol diagnostics.
3. Simulate common attack vectors (e.g., DoS, spoofing, ARP poisoning) and analyze their impact.
4. Reinforce concepts such as packet filtering, VPNs, IDS/IPS, and secure routing.
5. Prepare students to handle real-world security incidents and generate

response plans.

Course Outcomes (COs)

Upon successful completion of the lab, students will be able to:

1. Analyze network traffic using sniffing and packet inspection tools.
2. Configure firewall rules and access control policies in routers or VMs.
3. Simulate and detect common attacks (e.g., DoS, spoofing, DNS poisoning) in a test setup.
4. Implement VPNs and encrypted tunnels for secure data transfer.
5. Generate network security reports based on analysis and simulations.

Module A: Traffic Analysis and Sniffing

1. **Packet Capturing with Wireshark** – Analyze TCP three-way handshake, DNS, HTTP headers
2. **MAC and IP Spoofing Detection** – Use arping, ettercap, arpspoof
3. **Sniffing Login Credentials (Lab Sim)** – Use tcpdump or Wireshark to demonstrate plaintext risks
4. **Protocol Analysis Report** – Identify protocols and flags in captured traffic

Module B: Firewall, VPN, and Defense Tools

5. **Configuring iptables/ufw Firewall** – Block/unblock IPs, ports, and simulate rule layering
6. **Creating and Testing a VPN** – Use OpenVPN or WireGuard between virtual machines
7. **IDS/IPS with Snort or Suricata** – Basic rule configuration, detection of port scans
8. **Access Control via Router ACLs** – Simulated environment or Cisco Packet Tracer

Module C: Attack Simulation and Incident Handling

9. **Denial of Service Simulation (Test-Safe)** – Use hping3 or slowloris in a local lab
10. **DNS Spoofing Demonstration** – Use dnsspoof in isolated VM setup
11. **Brute-Force Attack Detection** – Use Hydra and monitor logs
12. **Incident Response Mini-Drill** – Analyze log dump, identify breach vector, write mini-report

Mini Project Ideas

- **Build and Defend a Secure Network** – Design a network topology with

layered defenses

- **Firewall Rule Optimizer** – Create a script to suggest ACL improvements
- **Attack Simulation Platform** – Configure dual-VM attacker/defender testing playground

Lab Tools & Environments

- **Wireshark**, **tcpdump**
- **iptables**, **ufw**, **pfSense**
- **OpenVPN**, **WireGuard**
- **Snort**, **Suricata**
- **Nmap**, **Zenmap**, **hping3**, **Hydra**
- **VirtualBox**, **VMware**, **Kali Linux**, **Ubuntu Server**

Textbooks and References

Lab References

1. **Chris Sanders** – ***Practical Packet Analysis***
2. **Mike Chapple** – ***CompTIA Security+ Guide to Network Security Fundamentals***
3. **Wireshark Official Lab Guide** – Free download from wireshark.org
4. **NIST SP 800-94** – Guide to Intrusion Detection and Prevention Systems

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Analyze network traffic using inspection tools	√	√	√	√	√				√	
CO2 – Configure firewall rules and ACLs	√	√	√	√	√	√			√	√
CO3 – Simulate and detect common attacks	√	√	√	√	√	√			√	√
CO4 – Implement VPNs and encrypted tunnels	√	√	√	√	√				√	√
CO5 – Generate network security reports	√			√	√	√	√	√	√	√

SEMESTER – II	
CSCP 460B	**ETHICAL HACKING LAB**

Course Objectives

This lab aims to:

1. Provide practical experience in ethical hacking using real-world tools and scenarios.
2. Simulate the phases of penetration testing in controlled environments.
3. Reinforce the principles of reconnaissance, vulnerability exploitation, and reporting.
4. Build familiarity with industry-grade tools such as Nmap, Metasploit, Burp Suite, and Wireshark.
5. Foster an ethical mindset while engaging in controlled attack simulations.

Course Outcomes (COs)

Upon completion of the lab, students will be able to:

1. Conduct footprinting and reconnaissance using OSINT tools.
2. Perform scanning, enumeration, and vulnerability identification on test systems.
3. Exploit known vulnerabilities and simulate attack chains.
4. Demonstrate post-exploitation techniques and cleanup procedures.
5. Prepare ethical hacking reports detailing findings, severity, and remediation.

Module A: Reconnaissance and Scanning

1. **Footprinting a Target** – Use WHOIS, nslookup, Google Dorking, Shodan
2. **Network Scanning with Nmap** – Discover hosts, ports, services
3. **Banner Grabbing and OS Fingerprinting** – With Telnet, Nmap
4. **DNS Enumeration and Subdomain Bruteforce** – With dnsenum, dirb

Module B: Exploitation and Vulnerability Testing

5. **Vulnerability Scanning with OpenVAS/Nessus** – Against test VM
6. **Brute-Force SSH/FTP with Hydra** – On lab-created users
7. **Web Exploits with OWASP Juice Shop** – SQL Injection, XSS, CSRF
8. **Using Metasploit to Exploit Known CVEs** – Against test machine (e.g., Windows XP, DVWA)

Module C: Post-Exploitation and Reporting

9. **Maintaining Access** – Creating reverse shells and backdoors (Netcat, msfvenom)

10. **Privilege Escalation Basics** – Local enumeration, kernel exploit concepts
11. **Log Manipulation and Cleanup** – Remove bash history, alter logs
12. **Writing a Penetration Testing Report** – Including screenshots, CVSS scores, and recommendations

• Mini Project Ideas

- **End-to-End PenTest Simulation** – Student scans, exploits, and reports on a purposely vulnerable VM (e.g., Metasploitable2)
- **Vulnerability Disclosure Report** – Based on ethical analysis of an open-source web app
- **Hacking Diary** – Document steps taken in multiple engagements, tools used, and defense strategies learned

Lab Tools & Platforms

- **Kali Linux**, **Parrot OS**
- **Metasploit**, **Burp Suite**, **Hydra**, **Nikto**, **Sqlmap**
- **OWASP Juice Shop**, **DVWA**, **Metasploitable2**, **Hack The Box (offline challenges)**
- **Nmap**, **Wireshark**, **Netcat**, **msfvenom**, **Armitage**

Textbooks and References

Lab References

1. **Georgia Weidman** – ***Penetration Testing: A Hands-On Introduction to Hacking***
2. **Vivek Ramachandran** – ***The Metasploit Megaprimer (Online Video Series)***
3. **EC-Council CEH v12 Practical Guidebook**
4. **Hack The Box and TryHackMe lab series**

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Conduct footprinting using OSINT	√	√	√	√	√				√	
CO2 – Perform scanning, enumeration, and vulnerability ID	√	√	√	√	√				√	
CO3 – Exploit vulnerabilities and simulate attack chains	√	√	√	√	√	√			√	√
CO4 – Demonstrate post-exploitation and cleanup	√	√	√	√	√	√			√	√

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO5 – Prepare ethical hacking reports	√			√	√	√	√	√	√	√

SEMESTER – II	
CSCO 462B	**CYBER AWARENESS & DIGITAL HYGIENE**

Course Objectives

This course aims to:

1. Create awareness about digital threats faced by individuals in everyday life.
2. Promote safe browsing, social media discipline, and device hygiene.
3. Introduce basic practices for protecting personal and professional information online.
4. Familiarize learners with phishing, scam tactics, and fraud prevention strategies.
5. Encourage a culture of responsible, ethical, and privacy-conscious digital behavior.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Identify common cyber threats like phishing, malware, and social engineering.
2. Practice safe browsing habits and password management techniques.
3. Protect digital identity and personal information on social media.
4. Apply basic device hygiene: updates, antivirus, backups, and safe app usage.
5. Report suspicious activity and practice informed digital citizenship.

UNIT I: Introduction to Digital Threats and Personal Risk *(12 Hours)*

Cybercrime in Daily Life, Phishing Emails and Fake Websites, Malware Types: Virus, Ransomware, Spyware, Safe Email and Messaging Practices, Financial Frauds: OTP Theft, QR Code Scam, Identity Theft Basics, Scenarios: Cyber Cafe Misuse, USB Risks, Mobile Scam Callers

Outcome: Awareness of real-life personal cyber threats and how to spot them.

UNIT II: Safe Practices and Cyber Hygiene *(12 Hours)*

Strong Password Practices and Password Managers, 2-Factor Authentication and OTP Safety, Software Updates and Patch Management, Mobile Device Hygiene: App Permissions, Data Backups, Antivirus Use, Secure Browsing: HTTPS, Ad Blockers, Avoiding Public Wi-Fi, Recognizing Fake News and Information Verification

Outcome: Learners apply digital hygiene habits across devices and platforms.

UNIT III: Digital Responsibility, Ethics, and Help Resources *(12 Hours)*

Cyber Etiquette and Digital Citizenship, Social Media Privacy Settings, Avoiding Online Harassment and Cyberbullying, Reporting Cybercrime: Cyber Cell, CERT-In, Helpline 1930, Cyber Laws Overview (India), IT Rules 2021 (Intro), Case Studies of Personal Mistakes and Lessons Learned

Outcome: Students become ethical, vigilant, and confident digital participants

Books & Supplementary Material

Textbooks / Guides

1. **Ministry of Electronics and IT (MeitY)** – ***Cyber Hygiene for Citizens*** (Free PDF)
2. **CERT-In Handbook** – ***Safe Online Practices for Students***
3. **Garima Tiwari** – ***Cyber Security Essentials for Everyday Users*** *(Beginner-level, Indian audience focused)*

Supplementary Resources

- https://cybercrime.gov.in – Official Government Portal
- https://www.cert-in.org.in – Indian CERT site
- https://stopthinkconnect.org – Global Cyber Hygiene Campaign
- Google Digital Citizenship Curriculum (Free)
- Cyber Hygiene YouTube Shorts / Public Info Ads

Optional Activities (Non-evaluative)

- **Phishing Spot-the-Fake Challenge**
- **Password Health Quiz**
- **Social Media Privacy Check Workshop**
- **Cyber Poster / Meme Contest**
- **Mini Skit on Cyberbullying Awareness**

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Identify common cyber	√	√			√	√				

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
threats										
CO2 – Practice safe browsing and password hygiene	✓	✓	✓		✓				✓	
CO3 – Protect digital identity on social media	✓				✓	✓	✓	✓		
CO4 – Apply basic device hygiene practices	✓	✓	✓	✓	✓				✓	✓
CO5 – Report suspicious activity, practice digital citizenship	✓			✓	✓	✓	✓	✓	✓	✓

SEMESTER – III

CSCH 501B	**CYBER THREAT INTELLIGENCE**

Course Objectives

This course aims to:

1. Introduce the principles and lifecycle of cyber threat intelligence.
2. Train students to collect, process, and analyze threat data from various sources.
3. Familiarize learners with CTI platforms, frameworks, and formats.
4. Enable correlation between IOCs, TTPs, and threat actors using structured techniques.
5. Prepare learners to use CTI for strategic, tactical, and operational defense decisions

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Explain the cyber threat intelligence lifecycle and its role in cybersecurity.
2. Collect and process Indicators of Compromise (IOCs) from open-source and internal feeds.
3. Analyze threat actor behavior using MITRE ATT&CK and kill chain models.
4. Utilize CTI platforms and formats like STIX, TAXII, and OpenCTI.
5. Correlate intelligence with real-world scenarios to support decision-making and risk reduction.

UNIT I: Introduction to Threat Intelligence and Lifecycle (12 Hours)

Definition and Types of Threat Intelligence, Strategic vs Tactical vs Operational Intelligence, Threat Intelligence Lifecycle: Planning, Collection, Processing, Analysis, Dissemination, Feedback, Intelligence Sources: OSINT, Dark Web, HUMINT, SIGINT, Threat Feeds (AlienVault, AbuseIPDB, MISP), Ethics of Intelligence Gathering

Outcome: Understand intelligence categories and operational pipeline.

UNIT II: Threat Actor Profiling and Analysis Techniques (12 Hours)

MITRE ATT&CK Framework, Lockheed Martin Kill Chain Model, Tactics, Techniques, and Procedures (TTPs), Indicators of Compromise (IOCs), Threat Actor Attribution and Campaign Tracking, Malware Families and Signature Behavior, Case Studies of APT Groups (APT28, Lazarus), Use of Threat Hunting for Early Detection

Outcome: Analyze adversary behavior and identify patterns in threats.

UNIT III: CTI Tools, Frameworks and Applications (12 Hours)

Structured Threat Intelligence Expression (STIX), Trusted Automated Exchange of Intelligence Information (TAXII), MISP and OpenCTI Platforms, Threat Intel Correlation with SIEM Tools, Threat Reports and Dashboards, Industry Use-Cases: CERT, SOCs, ISACs, CTI in Incident Response and Risk Prioritization, Limitations and Bias in CTI

Outcome: Use tools to interpret, organize, and act upon threat intelligence data.

Textbooks & Resources

Primary Textbooks

1. **Henry Dalziel – *Cyber Threat Intelligence* (Syngress)**
   ➤ Short, crisp, hands-on, very readable.
2. **Travis Rosiek – *Practical Cyber Threat Intelligence* (Packt)**
   ➤ A more hands-on practitioner's view.

Reference Books

- **Alex Walter – *Threat Intelligence Handbook* (Recorded Future)**
- **MITRE ATT&CK Navigator & GitHub**
- **ISACA – Cybersecurity Fundamentals Guide**
- **SANS Reading Room – Threat Intel Whitepapers**

Supplementary Resources

- https://attack.mitre.org – MITRE ATT&CK Matrix
- https://oasis-open.github.io/cti-documentation – STIX/TAXII official guide

- https://misp-project.org – Open source threat sharing platform
- Threat feeds: **AbuseIPDB, Anomali, AlienVault OTX, IBM X-Force Exchange**

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Explain the CTI lifecycle and its cybersecurity role	√				√	√		√		
CO2 – Collect and process IOCs from threat feeds	√	√	√	√	√				√	
CO3 – Analyze threat actors via ATT&CK and kill chain	√	√	√	√	√			√	√	√
CO4 – Utilize CTI platforms and data formats	√	√	√	√	√			√	√	√
CO5 – Correlate intelligence with real-world scenarios	√	√	√	√	√	√	√	√	√	√

SEMESTER – III	
CSCH 502B	**MALWARE ANALYSIS**

Course Objectives

This course aims to:

1. Introduce the types, behaviors, and structures of malware.
2. Train students in both static and dynamic analysis techniques.
3. Familiarize learners with tools used in reverse engineering, sandboxing, and disassembling.
4. Teach signature creation, behavior classification, and malware family identification.
5. Develop capability to document, report, and mitigate malware effectively.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Classify types of malware and understand their propagation and evasion strategies.
2. Perform static analysis using binary inspection and string extraction tools.
3. Conduct dynamic analysis in a controlled sandbox environment.
4. Use reverse engineering tools to deconstruct malware logic.

5. Generate reports and mitigation strategies for real-world malware samples.

UNIT I: Malware Fundamentals and Taxonomy (12 Hours)

Definition and History of Malware, Malware Classifications: Virus, Worm, Trojan, Rootkit, Keylogger, Ransomware, Spyware, Fileless Malware, Obfuscation and Packing Techniques, Malware Lifecycle and Infection Vectors, Persistence Mechanisms, Case Studies: Stuxnet, WannaCry, Pegasus, Redline

Outcome: Understand malware ecosystems and structural behaviors.

UNIT II: Static and Dynamic Analysis Techniques (12 Hours)

Static Analysis Concepts, File Fingerprinting (MD5, SHA256), Strings Analysis and Metadata Tools (strings, exiftool, PEiD), Dynamic Analysis Tools (Cuckoo Sandbox, Any.Run), API Call Monitoring, Registry and File Activity, Network Activity Inspection (Wireshark, TCPView), Malware Detonation in Isolated VMs

Outcome: Perform safe, observable malware behavior inspection and documentation.

UNIT III: Reverse Engineering and Reporting (12 Hours)

Assembly Basics (x86/x64), Disassemblers: IDA Pro, Ghidra, Binary Ninja (Intro), Debugging with OllyDbg/x64dbg, Analyzing Control Flow and Function Calls, Signature Creation and YARA Rules, Malware Family Classification, Reporting: TTP Documentation, CVE Referencing, IOC Extraction

Outcome: Learn to deconstruct and profile malware components and share findings professionally.

Textbooks & Resources

Primary Textbooks

1. **Michael Sikorski & Andrew Honig** – ***Practical Malware Analysis***
   No Starch Press – The classic, hands-on and comprehensive.
2. **Monnappa K A** – ***Learning Malware Analysis: Explore the Concepts, Tools, and Techniques***
   Packt Publishing – Beginner-friendly and India-oriented.

Reference Books & Tools

- **Tyler Hudak** – ***Reverse Engineering Malware*** (SANS)
- **YARA Documentation** – https://virustotal.github.io/yara/
- **Cuckoo Sandbox Docs** – https://cuckoosandbox.org
- **IDA Pro / Ghidra (NSA's Free Tool)**
- Malware samples: https://malware-traffic-analysis.net, [VirusShare], [Hybrid Analysis]

Supplementary Materials

- **VirusTotal Reports**
- **Any.run** – Interactive malware sandbox
- **FLARE VM** – Malware analyst's Windows toolkit
- **RE tools like PEStudio, Detect It Easy (DIE), CAPA**

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Classify malware types and strategies	✓				✓	✓				
CO2 – Perform static analysis with inspection tools	✓	✓	✓	✓	✓				✓	
CO3 – Conduct dynamic sandbox-based analysis	✓	✓	✓	✓	✓	✓			✓	✓
CO4 – Use reverse engineering to deconstruct malware	✓	✓	✓	✓	✓	✓			✓	✓
CO5 – Generate reports and recommend mitigation	✓	✓	✓	✓	✓	✓	✓	✓	✓	✓

SEMESTER – III

CSCH 503B	DATA MANAGEMENT & SECURITY

Course Objectives

This course aims to:

1. Introduce principles of data modeling, storage, and access in secure systems.
2. Explain vulnerabilities and threats related to data at rest, in use, and in transit.
3. Explore data security controls including encryption, masking, anonymization, and access control.
4. Teach database hardening and audit mechanisms.
5. Integrate compliance, governance, and privacy laws into the management of digital data.

Course Outcomes (COs)

Upon successful completion of the course, students will be able to:

1. Design secure data models and access patterns aligned with organizational needs.

2. Identify risks to data in different states and propose security controls.
3. Apply encryption, tokenization, and anonymization to protect sensitive data.
4. Evaluate policies for secure data lifecycle management and backup strategies.
5. Implement compliance-driven practices for secure and auditable data environments.

UNIT I: Data Fundamentals and Modeling for Security (12 Hours)

Data Lifecycle: Creation, Usage, Storage, Archival, Deletion, Types of Data: Structured, Unstructured, Semi-Structured, Data Models: Relational, Document, Columnar, Key-Value, Normalization vs Performance Trade-offs, Secure Schema Design, Roles and Access Mapping (RBAC, ABAC), Threats to Databases: Injection, Privilege Abuse, Misconfiguration

Outcome: Understand how structured planning reduces data vulnerability.

UNIT II: Data Protection Techniques and Architecture (12 Hours)

Encryption at Rest and In Transit, Transparent Data Encryption (TDE), Field-level vs Full-disk Encryption, Data Masking and Tokenization, Data Anonymization and Differential Privacy, Key Management Practices, Database Activity Monitoring (DAM), Logging and Change Tracking, RAID and Backup Best Practices

Outcome: Apply protection measures across data states using modern tools and methods.

UNIT III: Governance, Compliance, and Secure Data Operations (12 Hours)

Data Privacy Laws: GDPR, PDPB (India), HIPAA, Data Classification Policies, Secure Data Disposal and Retention Schedules, Database Auditing and Alerts, Compliance Reporting, Cloud Data Security Principles, Multi-tenancy and Tenant Isolation, Data Sharing Agreements and DLP Policies, Secure Data Engineering Culture

Outcome: Align data practices with legal, ethical, and operational expectations.

Textbooks & Resources

Primary Textbooks

1. **Michael Gertz & Sushil Jajodia** – ***Handbook of Database Security***
   Springer – Comprehensive and academically rich.
2. **Pierangela Samarati et al.** – ***Data Security and Privacy in Cloud Computing***
   Springer Briefs – Great coverage of anonymization and cloud storage risks.

Reference Books

- **Paul C. Zikopoulos** – ***Understanding Big Data Security***
- **Siberschatz, Korth** – ***Database System Concepts*** (For schema & theory background)
- **O'Reilly** – ***Data Management at Scale*** **(Architectural perspective)**
- **Oracle / MS SQL Security Best Practices Guides**

Supplementary Resources

- https://gdpr.eu – Complete GDPR resource
- https://meity.gov.in – India's Data Protection Bill updates
- https://differentialprivacy.org – for anonymization research
- Tools: **Vormetric, IBM Guardium, AWS KMS, VeraCrypt, TDE setups in MySQL/PostgreSQL**

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Design secure data models and access patterns	√	√	√	√	√				√	√
CO2 – Identify risks and propose security controls	√	√	√	√	√	√			√	√
CO3 – Apply encryption, tokenization, anonymization	√	√	√	√	√	√			√	√
CO4 – Evaluate secure lifecycle and backup strategies	√	√	√	√	√	√	√		√	√
CO5 – Implement compliance-driven and auditable practices	√	√	√	√	√	√	√	√	√	√

SEMESTER – III

CSCS 504B	DIGITAL FORENSICS

Course Objectives

This course aims to:

1. Introduce students to the fundamental principles and process of digital forensic investigations.
2. Provide an understanding of evidence acquisition, preservation, and analysis across platforms.
3. Train students on forensic tools for analyzing storage media, memory, logs, and networks.
4. Emphasize chain-of-custody, legal admissibility, and report writing.

5. Enable students to perform forensic reconstruction of incidents in both desktop and mobile contexts.

Course Outcomes (COs)

By the end of the course, students will be able to:

1. Describe the digital forensic process and legal requirements for admissibility.
2. Perform imaging and preservation of digital evidence using forensic tools.
3. Analyze file systems, logs, and memory dumps for forensic artifacts.
4. Reconstruct events from browser history, email, metadata, and timestamps.
5. Document and present a forensically sound report respecting chain-of-custody standards.

UNIT I: Introduction to Digital Forensics and Evidence Handling (12 Hours)

Principles of Digital Forensics, Types of Digital Evidence: Volatile and Non-Volatile, Phases of Investigation: Identification, Collection, Examination, Analysis, Presentation, Legal Considerations and IT Act Provisions, Chain-of-Custody Documentation, Write Blockers, Imaging Tools and Hash Verification (MD5, SHA), Role of Forensic Labs and Cyber Cells

Outcome: Students understand the scope and sensitivity of forensic investigations.

UNIT II: Forensic Analysis of Devices and Artifacts (12 Hours)

File System Forensics: FAT, NTFS, ext, slack space analysis, Registry Forensics, Browser and Email Artifact Recovery, USB Usage Traces, Deleted File Recovery Techniques, RAM and Pagefile Analysis, Keyword Searching and Timeline Creation, Mobile Forensics Basics (Android & iOS), Metadata and EXIF Analysis

Outcome: Learners gain hands-on techniques for exploring device history and user actions.

UNIT III: Network, Malware, and Reporting Forensics (12 Hours)

Live Network Forensics, Packet Capture Analysis using Wireshark and NetworkMiner, Firewall and Router Log Analysis, Email Header Tracing, Malware Dropper and Payload Inspection, Correlating Forensic Artifacts to Attack Vectors, Forensic Report Writing Structure, Legal Admissibility and Expert Testimony Basics, Case Studies of Forensic Investigations

Outcome: Students synthesize findings into formal reports and support legal conclusions.

Textbooks & Resources

Primary Textbooks

1. **Nelson, Phillips, Enfinger – *Guide to Computer Forensics and Investigations* (Cengage)**
   Widely adopted and hands-on.
2. **John Sammons – *The Basics of Digital Forensics* (Syngress)**
   Concise and beginner-friendly.

Reference Books

- **Eoghan Casey – *Digital Evidence and Computer Crime***
- **Harlan Carvey – *Windows Forensic Analysis Toolkit***
- **Brian Carrier – *File System Forensic Analysis*** (For FAT, NTFS deep dives)

Supplementary Tools & Platforms

- **Autopsy (Sleuth Kit GUI)**
- **FTK Imager**, **X-Ways Forensics**
- **Volatility Framework** – Memory analysis
- **Wireshark**, **NetworkMiner** – Network artifacts
- **ExifTool**, **Bulk Extractor**, **Magnet AXIOM** (Demo)
- https://www.digitalforensicsmagazine.com
- https://hashlookup.circl.lu – Known hash database

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Describe forensic process and legal standards	√				√	√	√	√		√
CO2 – Perform imaging and evidence preservation	√	√	√	√	√				√	√
CO3 – Analyze file systems, logs, and memory dumps	√	√	√	√	√				√	√
CO4 – Reconstruct events from digital artifacts	√	√	√	√	√			√	√	√
CO5 – Document and present a forensically sound report	√	√	√	√	√	√	√	√	√	√

SEMESTER – III	
CSCS 505B	**WEB DEVELOPMENT & SECURE CODING**

Course Objectives

This course aims to:

1. Teach the fundamentals of web development using a full-stack approach.
2. Introduce common web vulnerabilities and their exploitation techniques.
3. Emphasize secure design principles and secure coding guidelines.
4. Equip students with best practices for authentication, session handling, and input validation.
5. Prepare learners to build defensible, maintainable, and standards-compliant web applications.

Course Outcomes (COs)

By the end of the course, students will be able to:

1. Design and develop secure web applications using front-end and back-end technologies.
2. Identify and remediate common web application vulnerabilities.
3. Implement secure authentication, session management, and access control.
4. Apply input validation, encoding, and secure error handling practices.
5. Deploy applications with logging, monitoring, and secure development lifecycle awareness.

UNIT I: Full-Stack Web Development Basics (12 Hours)

HTTP Protocol and REST Principles, Frontend Basics: HTML, CSS, JavaScript, DOM Manipulation, Backend with Python (Flask or Django), Routing and Templates, Database Integration with SQLite/PostgreSQL, Client–Server Communication (AJAX, Fetch), JSON Data Handling, Hosting Basics (localhost, cloud deploy)

Outcome: Students can build functional web apps using open-source stacks.

UNIT II: Web Vulnerabilities and Secure Coding Practices (12 Hours)

OWASP Top 10: SQL Injection, XSS, CSRF, Broken Authentication, Security Misconfiguration, Input Validation Techniques, Output Encoding, Secure File Uploads, Error Handling and Logging Practices, Session Fixation and Cookie Protection, Secure Authentication with JWT/OAuth

Outcome: Students can identify and mitigate common web threats during development.

UNIT III: Secure Design, Deployment, and Testing (12 Hours)

Principles of Secure Software Design, Security Headers (CSP, X-Frame-Options, etc.), TLS and HTTPS Enforcement, Secure API Development, Code Review and Static Analysis Tools (Bandit, SonarQube), Web Application Firewalls (Intro), Logging and Audit Trails, Deployment Hardening, Secure DevOps Awareness

Outcome: Students learn to plan, test, and deploy production-grade secure systems.

Textbooks & Resources

Primary Textbooks

1. **Bryson Payne** – ***Learn JavaScript with Python for Web Development and Security***
   No Starch Press – Ideal for secure full-stack focus.
2. **Bryan Sullivan & Vincent Liu** – ***Web Application Security: A Beginner's Guide***
   McGraw-Hill

Reference Books

- **OWASP Developer Guide** (Free, evolving best-practices)
- **Joel Scambray** – ***Hacking Exposed: Web Applications***
- **Michael Howard** – ***Writing Secure Code*** **(Microsoft Press)**
- **Miguel Grinberg** – ***Flask Web Development*** **(O'Reilly)**
- **FreeCodeCamp.org** – Full-stack crash courses

Supplementary Tools & Platforms

- **DVWA**, **OWASP Juice Shop** – Secure coding playgrounds
- **SonarQube**, **Bandit**, **Brakeman** – Static analysis
- **Burp Suite**, **Postman**, **Zap Proxy** – Security testing
- https://owasp.org – Essential reading for every secure coder
- https://portswigger.net/web-security – Interactive learning

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Design and develop secure web applications	√	√	√	√	√				√	√
CO2 – Identify and remediate web vulnerabilities	√	√	√	√	√				√	√
CO3 – Implement secure auth, session, and access control	√	√	√	√	√	√			√	√
CO4 – Apply input validation, encoding, error handling	√	√	√	√	√				√	√
CO5 – Deploy with logging, monitoring, and SDLC awareness	√	√	√	√	√	√	√	√	√	√

SEMESTER – III

CSCP 509B	CYBER THREAT INTELLIGENCE LAB

Course Objectives

This lab aims to:

1. Provide hands-on experience in collecting, analyzing, and acting on threat intelligence.
2. Introduce industry-standard tools and platforms used in CTI workflows.
3. Train students to extract Indicators of Compromise (IOCs) and use MITRE ATT&CK for profiling.
4. Foster analytical thinking for correlating threats, behaviors, and campaign patterns.
5. Build capabilities in producing actionable threat intelligence reports.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Collect and process open-source threat intelligence using OSINT tools.
2. Extract, validate, and enrich IOCs from threat feeds and malware reports.
3. Analyze adversary behavior using MITRE ATT&CK Navigator.
4. Use MISP/OpenCTI to share, correlate, and visualize CTI data.
5. Prepare and present actionable intelligence in technical report format.

Module A: Threat Data Collection & IOC Extraction

1. **Using OSINT Tools (theHarvester, Shodan, SpiderFoot)** – Collect domains, IPs, email leaks
2. **Parsing Threat Reports (FireEye, CrowdStrike, Recorded Future)** – Extract IOCs manually
3. **IOC Format Conversion** – STIX to plain text, CSV to JSON
4. **Hash and Domain Lookup via VirusTotal API** – Validate known bad indicators

Module B: Behavioral and TTP Analysis

5. **TTP Mapping with MITRE ATT&CK Navigator** – Tag observed behaviors
6. **Threat Actor Profiling** – Research and profile APT groups from ATT&CK/FireEye
7. **Timeline Creation** – Use IOC timestamps to reconstruct breach narratives
8. **Kill Chain Analysis** – Apply Lockheed Martin model to real-world APT case

Module C: CTI Platforms and Reporting

9. **MISP Platform Usage** – Create events, tag indicators, share with groups
10. **TAXII Feed Consumption** – Ingest threat data into MISP/OpenCTI
11. **Correlation Drill** – Link malware family, IPs, and actor based on feeds
12. **Mini CTI Report Writing** – Include severity, scope, and suggested response

Tools & Platforms

- **MISP (Malware Information Sharing Platform)**
- **MITRE ATT&CK Navigator**
- **OpenCTI (Threat Intelligence Platform)**
- **VirusTotal**, **AbuseIPDB**, **Shodan**, **AlienVault OTX**
- **CyberChef**, **IOC Parser**, **YETI CTI Workbench (optional)**

Reference Materials

1. **Threat Intelligence Handbook – Recorded Future**
2. **MISP Training Docs – misp-project.org**
3. **MITRE ATT&CK Official Wiki – attack.mitre.org**
4. **Alex Walter – *Cyber Threat Intelligence Essentials***
5. SANS Whitepapers on Threat Hunting and CTI

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Collect and process OSINT data	√	√	√	√	√				√	
CO2 – Extract, validate, and enrich IOCs	√	√	√	√	√				√	√
CO3 – Analyze adversary behavior with MITRE ATT&CK	√	√	√	√	√			√	√	√
CO4 – Use MISP/OpenCTI for CTI data operations	√	√	√	√	√	√		√	√	√
CO5 – Prepare and present actionable CTI reports	√	√	√	√	√	√	√	√	√	√

SEMESTER – III	
CSCP 510B	**MALWARE ANALYSIS LAB**

Course Objectives

This lab aims to:

1. Provide hands-on experience with live malware in controlled, virtual environments.
2. Train students in static and dynamic analysis methods.
3. Familiarize learners with real-world malware samples, obfuscation, and behavior analysis.
4. Develop skill in signature generation, YARA rule creation, and IOC extraction.
5. Instill safety practices and forensic discipline while handling digital malware.

Course Outcomes (COs)

Upon successful completion of the lab, students will be able to:

1. Safely isolate and analyze malicious files using static and dynamic methods.
2. Monitor malware behavior through system calls, registry changes, and network traffic.
3. Use reverse engineering tools to disassemble or debug malware binaries.
4. Create custom signatures for detection using YARA and hash-based methods.
5. Document malware functionality and recommend mitigation strategies.

Module A: Static Analysis and Fingerprinting

1. **Hashing and Metadata Extraction** – Use md5sum, sha256sum, exiftool
2. **Strings and PE Header Inspection** – strings, PEStudio, die (Detect It Easy)
3. **File Type Analysis and Entropy Check** – Packed/Obfuscated detection
4. **Manual IOC Extraction from Reports** – Hashes, IPs, registry keys, mutexes

Module B: Dynamic Analysis and Behavior Monitoring

5. **Using Cuckoo Sandbox** – Malware detonation and behavior capture
6. **Wireshark for Network Behavior** – DNS, HTTP, and suspicious connections
7. **Registry and File Monitoring Tools** – Regshot, ProcMon, Process Hacker
8. **Basic Script-Based Malware Execution** – Analysis of .vbs/.bat droppers

Module C: Reverse Engineering and Reporting

9. **Disassembly with Ghidra or IDA Free** – Analyze functions and control flow
10. **Debugging with x64dbg/OllyDbg** – Track execution, set breakpoints

11. **Writing Basic YARA Rules** – Signature-based detection patterns
12. **Malware Behavior Report** – Structure: summary, persistence, payload, detection

Tools & Platforms

- **Cuckoo Sandbox**, **Wireshark**, **Process Monitor**
- **PEStudio**, **YARA**, **Ghidra**, **x64dbg**, **Volatility (for memory samples)**
- **DIE (Detect It Easy)**, **Any.run (Optional Cloud Sandbox)**
- Malware samples from **malware-traffic-analysis.net**, **VirusShare**, **Hybrid Analysis**

Reference Materials

1. **Michael Sikorski & Andrew Honig** – ***Practical Malware Analysis***
2. **Monnappa K A** – ***Learning Malware Analysis***
3. **Ghidra User Guide & Malware Playbook (FireEye)**
4. **YARA Documentation** – **https://virustotal.github.io/yara/**
5. **REmap (Reverse Engineering Resources)** by Malware Unicorn

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Isolate and analyze malicious files	√	√	√	√	√				√	√
CO2 – Monitor malware behavior via system and network activity	√	√	√	√	√				√	√
CO3 – Disassemble/debug malware binaries	√	√	√	√	√	√			√	√
CO4 – Create custom YARA/hash-based signatures	√	√	√	√	√				√	√
CO5 – Document functionality and recommend mitigation	√	√	√	√	√	√	√	√	√	√

SEMESTER – III

CSCO 512B	CRIMES, CYBERCRIME, DIGITAL EVIDENCE AND CYBER LAW

Course Objectives

This course aims to:

1. Introduce students to the legal foundations of crime and cybercrime.
2. Clarify how traditional laws adapt (or fail) in cyberspace.
3. Explain the collection, preservation, and admissibility of digital evidence.
4. Develop awareness of cybercrime categories, from fraud to cyberbullying.
5. Foster ethical, lawful, and responsible digital citizenship.

Course Outcomes (COs)

Upon completion of this course, students will be able to:

1. Differentiate between conventional crimes and cybercrimes in legal terms.
2. Identify key legal provisions in Indian cyber law, including IT Act 2000.
3. Understand the nature, collection, and admissibility of digital evidence.
4. Recognize personal and organizational responsibilities under cyber law.
5. Reflect on ethical, social, and legal implications of actions in digital space.

UNIT I: Introduction to Crime, Cybercrime and Jurisprudence (12 Hours)

Definition of Crime and Classification, Cybercrime vs Traditional Crime, Evolution of Cybercrime, Elements of Crime: Actus Reus and Mens Rea, Jurisdiction Challenges in Cyberspace, Categories of Cybercrime: Financial, Hate Speech, Terrorism, Defamation, Identity Theft, Revenge Porn, Cyberstalking

Outcome: Students grasp the shifting landscape of criminality in the digital era.

UNIT II: Digital Evidence and Investigation Process (12 Hours)

Types of Digital Evidence: Volatile and Non-Volatile, Principles of Evidence Collection, Hashing and Chain of Custody, Admissibility of Digital Evidence in Indian Courts, Search and Seizure in Digital Context, Role of CERT-In and Law Enforcement, Case Studies: Email Tracing, Device Imaging, Mobile Seizure

Outcome: Students understand what makes digital evidence forensically and legally valid.

UNIT III: Cyber Law and Legal Frameworks in India (12 Hours)

Information Technology Act 2000 and Amendments, Section 43, 66, 67, 72 and 79 of IT Act, Intermediary Guidelines, Role of Adjudicating Officers and Cyber Appellate Tribunal, Data Protection (PDPB), Comparison with GDPR, Cyber Law and Ethics, Reporting Cybercrime: Portals, FIR, and Legal Remedies

Outcome: Students will recognize the powers and limits of cyber law in digital governance.

Textbooks & Resources

Primary Textbooks

1. **Pavan Duggal** – ***Cyber Law: The Indian Perspective***
   (Universal Law Publishing)
2. **Talat Fatima** – ***Cyber Crimes***
   (LexisNexis India)

Reference Books

- **Rodney Ryder** – ***Guide to Cyber Laws***
- **Justice Yatindra Singh** – ***Cyber Laws***
- **Sanjay Pandey** – ***Indian Cyber Law***
- **Eoghan Casey** – ***Digital Evidence and Computer Crime*** (Forensics + Law bridge)

Supplementary Platforms

- https://cybercrime.gov.in – Indian Cybercrime Portal
- https://meity.gov.in – Ministry of Electronics & IT
- https://cert-in.org.in – National incident response
- Indian Kanoon: Browse judgments involving cybercrime
- **YouTube Channels**: NALSAR, PRS India Cyberlaw Talks, LegalEdge (cyber focus)

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Differentiate between conventional and cybercrimes	√				√	√	√			
CO2 – Identify key provisions in Indian cyber law	√				√	√	√	√		√
CO3 – Understand digital evidence and admissibility	√	√			√	√	√	√	√	√
CO4 – Recognize responsibilities under cyber law	√	√		√	√	√	√	√	√	√
CO5 – Reflect on ethical, social, legal implications	√				√	√	√	√	√	√

SEMESTER – IV

CSCH 551B	APPLICATION SECURITY

Course Objectives

This course aims to:

1. Provide in-depth understanding of vulnerabilities across application layers.
2. Introduce secure software development lifecycle (SSDLC) and shift-left security principles.
3. Equip students to detect, exploit, and remediate application-level flaws.
4. Familiarize learners with automated scanning tools, threat modeling, and mitigation patterns.
5. Align application design with OWASP Top 10 and DevSecOps best practices.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Identify security risks in client-side, server-side, and API interactions.
2. Analyze code and architecture to locate vulnerabilities and misconfigurations.
3. Apply principles of input validation, secure authentication, and output encoding.
4. Integrate security testing in SDLC using tools like SAST and DAST.
5. Recommend and implement mitigation strategies aligned with application threat models.

UNIT I: Introduction to Application Security and OWASP Top 10 (12 Hours)

Need for Application Security, Risk and Vulnerability Classifications, OWASP Top 10: Injection, Broken Authentication, Sensitive Data Exposure, XML External Entities (XXE), Security Misconfigurations, Cross-Site Scripting (XSS), Broken Access Control, Insecure Deserialization, Vulnerable Components, Insufficient Logging and Monitoring

Outcome: Students gain deep insight into the ten most critical application flaws.

UNIT II: Secure Software Development Lifecycle and Threat Modeling (12 Hours)

SSDLC Phases and Integration Points, Secure Design Principles (Least Privilege, Defense in Depth), STRIDE and DREAD Threat Modeling, Threat Mapping and Abuse Cases, Static and Dynamic Application Security Testing (SAST, DAST), Secure Code Review Patterns, Security Headers and Configuration Hardening, CI/CD Security Touchpoints

Outcome: Learners understand how security is embedded into the development pipeline.

UNIT III: Advanced Security Practices and Secure APIs (12 Hours)

Authentication Protocols (OAuth 2.0, JWT, SAML), Session Management Techniques, Input Sanitization and Output Encoding, Rate Limiting and Throttling, Secure API Design and Gateway Controls, SSRF, CSRF and CORS Misuse, AppSec in Microservices and Containers, Secure Logging and Incident Response, AppSec Case Studies: GitHub, Equifax, Log4Shell

Outcome: Students can design, build, and troubleshoot secure application ecosystems.

Textbooks & Resources

Primary Textbooks

1. **Andrew Hoffman – *Web Application Security: Exploitation and Countermeasures for Modern Web Apps***
   (O'Reilly)
2. **Bryan Sullivan & Vincent Liu – *Web Application Security: A Beginner's Guide***
   (McGraw-Hill)

Reference Books

- **OWASP Developer Guide & Cheat Sheets**
- **Michael Howard – *Writing Secure Code* (Microsoft Press)**
- **Neil Daswani – *Foundations of Security* (for software engineering tie-ins)**
- **OWASP Application Security Verification Standard (ASVS)** – Free online

Supplementary Tools & Platforms

- **Burp Suite**, **OWASP ZAP**, **SonarQube**, **Snyk**, **Bandit**
- **Github CodeQL**, **Brakeman (Ruby)**, **Semgrep**
- https://owasp.org – OWASP Top 10, ASVS, Cheat Sheets
- https://portswigger.net/web-security – Labs

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Identify security risks in various components	√	√	√	√	√				√	
CO2 – Analyze code and architecture for vulnerabilities	√	√	√	√	√				√	√
CO3 – Apply secure coding principles	√	√	√	√	√				√	√
CO4 – Integrate security testing into SDLC	√	√	√	√	√	√		√	√	√
CO5 – Recommend and implement mitigation strategies	√	√	√	√	√	√	√	√	√	√

SEMESTER – IV	
CSCH 552B	**INCIDENT RESPONSE & SECURITY OPERATIONS**

Course Objectives

This course aims to:

1. Introduce the lifecycle and methodology of incident response (IR).
2. Familiarize learners with real-time security operations and SOC workflows.
3. Train students to handle alerts, triage incidents, and investigate system anomalies.
4. Teach tools, automation, and SIEM platforms used in live security monitoring.
5. Develop strategic thinking in containment, recovery, and post-incident analysis.

Course Outcomes (COs)

Upon successful completion, students will be able to:

1. Describe the phases and framework of incident response lifecycle.
2. Triage, categorize, and escalate incidents based on severity and impact.
3. Use SIEM tools to monitor, correlate, and analyze security events.
4. Coordinate containment, eradication, and recovery strategies effectively.
5. Document incidents with timelines, artifacts, and lessons learned for IR playbooks.

UNIT I: Incident Response Lifecycle and Frameworks (12 Hours)

Definition of Incident and Event, Types of Security Incidents, NIST SP 800-61 IR Framework, IR Phases: Preparation, Identification, Containment, Eradication, Recovery, Lessons Learned, MITRE D3FEND, IR Playbooks, Roles of IR Team: SOC Analyst, IR Lead, Forensics, Communication Flow in Crisis

Outcome: Students understand structure, phases, and actor roles in IR strategy.

UNIT II: Security Operations and Monitoring Tools (12 Hours)

Security Operations Center (SOC) Overview, Tier I to Tier III SOC Roles, SIEM Tools: Splunk, ELK, QRadar, Event and Log Sources: Firewall, Endpoint, Authentication, Threat Intelligence Integration into SIEM, Alert Fatigue and Rule Tuning, IOC and Correlation Rule Design, Ticketing and Escalation Workflows

Outcome: Learners gain hands-on understanding of SOC operations and alert management.

UNIT III: Incident Handling and Post-Mortem Analysis (12 Hours)

Containment Strategies: Host, Network, Cloud, Memory Acquisition and Preservation, Root Cause Identification and Forensics, Recovery Metrics: RTO and RPO, Incident Timelining, Evidence Chain and Documentation, Report Writing and Stakeholder Communication, Post-Incident Review and Security Hardening, Case Studies: SolarWinds, Capital One, Colonial Pipeline

Outcome: Students learn to handle incidents end-to-end, with forensic and strategic rigor.

Textbooks & Resources

Primary Textbooks

1. **Jason Luttgens, Matthew Pepe, Kevin Mandia – *Incident Response & Computer Forensics* (McGraw-Hill)**
   ➤ Industry favorite, hands-on and highly relevant.
2. **Jorge Orchilles – *The Blue Team Field Manual (BTFM)***
   ➤ Practical quick-reference guide for IR teams.

Reference Books & Playbooks

- **NIST SP 800-61** – *Computer Security Incident Handling Guide*
- **MITRE D3FEND & ATT&CK Navigator**
- **LogRhythm SOC Analyst Guide**
- **CISA IR Playbooks** – https://www.cisa.gov

Supplementary Tools & Platforms

- **SIEM**: Splunk, ELK Stack, QRadar (demo), Wazuh
- **EDR**: Velociraptor, Sysmon + Windows Event Logs
- **Ticketing Tools**: TheHive, RTIR, Jira (for IR workflows)
- **Timeline Creators**: Timesketch, Plaso, CyberChef
- **Memory Capture**: FTK Imager, Volatility
- **IR Labs**: CyberDefenders, BlueTeamLabs, TryHackMe (Blue Side rooms)

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Describe IR lifecycle and framework	√				√	√		√		√
CO2 – Triage, categorize, and escalate incidents	√	√	√	√	√	√			√	√

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO3 – Use SIEM tools for event monitoring and analysis	√	√	√	√	√				√	√
CO4 – Coordinate containment and recovery strategies	√	√	√	√	√	√		√	√	√
CO5 – Document incidents and lessons for playbooks	√	√	√	√	√	√	√	√	√	√

SEMESTER – IV	
CSCH 553B	**AI IN CYBERSECURITY**

Course Objectives

This course aims to:

1. Introduce the role of AI and ML techniques in cyber defense.
2. Train students in designing models for anomaly detection, threat hunting, and malware classification.
3. Analyze strengths, limits, and ethics of AI use in cybersecurity operations.
4. Explore AI use cases in threat intelligence, phishing detection, SOC automation, and UEBA.
5. Build foundational skill in applying ML algorithms to security datasets using open-source libraries.

Course Outcomes (COs)

By the end of the course, students will be able to:

1. Explain the landscape and motivation for using AI/ML in cybersecurity.
2. Apply supervised and unsupervised techniques to analyze cyber threats.
3. Evaluate datasets using feature engineering and model validation techniques.
4. Build ML models for intrusion detection, spam/phishing filtering, and malware analysis.
5. Critically analyze ethical implications and adversarial risks in AI-enabled security systems.

UNIT I: AI and Machine Learning for Cyber Defense (16 Hours)

Introduction to AI/ML in Cybersecurity, Cybersecurity Data Types and Challenges, Supervised vs Unsupervised Learning, Feature Engineering and Normalization, Datasets: CICIDS, UNSW-NB15, NSL-KDD, Model Evaluation: Accuracy, Precision,

ROC Curve, Confusion Matrix, Basic ML Models: Decision Trees, KNN, Naive Bayes, SVM, Logistic Regression

Outcome: Students understand how ML can classify or cluster cyber threats.

UNIT II: Use Cases and Model Applications (16 Hours)

Intrusion Detection Systems (IDS) using AI, Anomaly Detection using Clustering and Autoencoders, Spam and Phishing Detection via NLP, Malware Classification from Opcode/API features, Behavioral Biometrics and UEBA, Threat Intelligence Enrichment, SOC Automation with ML Models, Model Deployment Basics using Flask/Streamlit

Outcome: Learners build and interpret real-world models across cyber domains.

UNIT III: Limitations, Ethics, and Future of AI in Security (16 Hours)

Adversarial Machine Learning and Model Poisoning, Explainable AI in Security, Model Drift and Retraining Strategies, Privacy-Preserving ML (Federated Learning, DP), Bias and False Positives in Security AI, Ethics in Automated Decision Making, AI for Red Team vs Blue Team, Future Trends: LLMs in SOC, AI-Augmented Analysts

Outcome: Students recognize boundaries, risks, and futures of intelligent defense.

Textbooks & Resources

Primary Textbooks

1. **Chet Hosmer** – ***Artificial Intelligence and Machine Learning for Cybersecurity***
   (Apress)
   ➤ Practical, Python-based, industry-focused.
2. **Soma Halder & Sinan Ozdemir** – ***Hands-On Machine Learning for Cybersecurity***
   (Packt Publishing)

Reference Materials

- **NIST IR 8286A – Use of AI in Security**
- **MITRE ATLAS** – Adversarial Threat Landscape for AI Systems
- **Google AI/ML Guides** – Best practices
- **UNSW-NB15 & CICIDS2017 Dataset Docs**
- **IEEE Xplore** – Research on AI in SOCs

Tools & Platforms

- **Scikit-learn**, **Pandas**, **Seaborn** – For modeling and EDA

- **TensorFlow/Keras** or **PyTorch** – For neural approaches
- **Snorkel**, **AutoML**, **HuggingFace** – Advanced/Optional
- **Streamlit/Flask** – For model deployment demos
- TryHackMe: AI in SOCs Labs (Blue team AI)

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Explain AI/ML landscape in cybersecurity	√				√	√	√	√		√
CO2 – Apply supervised and unsupervised learning	√	√	√	√	√				√	√
CO3 – Evaluate datasets and validate models	√	√	√	√	√				√	√
CO4 – Build ML models for cyber threat scenarios	√	√	√	√	√	√		√	√	√
CO5 – Analyze ethical and adversarial issues	√	√		√	√	√	√	√	√	√

SEMESTER – IV

CSCS 554B	BLOCKCHAIN, SECURITY & USE CASES

Course Objectives

This course aims to:

1. Explain the core principles, architecture, and consensus mechanisms of blockchain.
2. Analyze blockchain security features and common vulnerabilities.
3. Introduce smart contracts and decentralized application (dApp) fundamentals.
4. Explore major blockchain platforms and protocols through practical use cases.
5. Critically evaluate blockchain adoption in finance, identity, supply chain, and cybersecurity.

Course Outcomes (COs)

Upon completion of the course, students will be able to:

1. Explain blockchain structure, components, and consensus mechanisms.
2. Identify security strengths and weaknesses in public and private

blockchains.

3. Develop basic smart contracts and test them on a blockchain emulator.
4. Analyze real-world blockchain applications with technical and strategic depth.
5. Evaluate trade-offs and threats in adopting blockchain for cybersecurity and digital trust.

UNIT I: Blockchain Foundations and Architecture (12 Hours)

Blockchain Basics: Blocks, Hashes, Merkle Trees, Consensus Mechanisms: PoW, PoS, PBFT, Mining and Difficulty Adjustment, Blockchain Types: Public, Private, Consortium, Cryptographic Concepts: Digital Signatures, Hashing, Pseudonymity, Blockchain Trilemma: Scalability, Security, Decentralization, Forks and Chain Finality

Outcome: Students grasp blockchain as a tamper-resistant, distributed record system.

UNIT II: Blockchain Security and Smart Contracts (12 Hours)

Blockchain Security Features: Immutability, Non-repudiation, Privacy vs Transparency, Smart Contracts: Introduction, Design and Deployment, Ethereum and Solidity Basics, Common Attacks: Reentrancy, Timestamp Dependency, Gas Limit Exploits, Wallet Security and Key Management, Secure Coding for Smart Contracts

Outcome: Learners explore vulnerabilities and protections in on-chain logic and assets.

UNIT III: Use Cases and Future Applications (12 Hours)

Blockchain in Identity Management and Authentication, Supply Chain Transparency, Voting Systems and E-Governance, Blockchain in Finance (DeFi, CBDCs), Blockchain in Cybersecurity: Tamper-proof Logs, Threat Sharing, Blockchain vs Traditional Databases, Regulatory and Ethical Aspects, Web3 and Zero Trust Models, Interoperability and Scalability Challenges

Outcome: Students evaluate where blockchain fits (or doesn't) in solving real-world problems.

Textbooks & Resources

Primary Textbooks

1. **Arvind Narayanan et al. – *Bitcoin and Cryptocurrency Technologies*** *(Princeton University Press)*
   ➤ Strong theory with applied flavor.
2. **Imran Bashir – *Mastering Blockchain* (Packt)**
   ➤ Covers security, development, and advanced architectures.

Reference Books

- **Antonopoulos – *Mastering Ethereum* (O'Reilly)**
- **Joseph Bonneau – *SoK: Security of Blockchain Systems***
- **IEEE Blockchain Technical Briefs and NIST Blockchain Publications**

Tools & Supplementary Platforms

- **Remix IDE (Solidity)** – Smart contract development
- **Ganache** + **MetaMask** – Local blockchain testing
- **Truffle Suite** – DApp framework
- **Etherscan**, **Infura**, **Chainlink Docs** – Real-world contract data
- **TryHackMe: Smart Contract Exploitation**, **Ethernaut Game** – Interactive labs

CO \ PO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1 – Explain blockchain structure and consensus	√				√					
CO2 – Identify security strengths and weaknesses	√	√	√	√	√	√			√	√
CO3 – Develop and test smart contracts	√	√	√	√	√				√	√
CO4 – Analyze real-world blockchain use cases	√	√	√	√	√	√	√	√	√	√
CO5 – Evaluate adoption trade-offs and risks	√	√	√	√	√	√	√	√	√	√

SEMESTER – IV	
CSCS 555B	**IT AUDIT**

Course Objectives

This course aims to:

1. Introduce the principles, standards, and process of IT auditing.
2. Explain risk-based audit planning and control evaluation techniques.
3. Equip students to audit networks, applications, and information systems.
4. Familiarize learners with regulatory compliance (e.g., ISO 27001, COBIT, PCI-DSS).
5. Train students in audit documentation, reporting, and communication of

findings.

Course Outcomes (COs)

Upon completion, students will be able to:

1. Describe audit frameworks and lifecycle in IT and cybersecurity domains.
2. Conduct risk assessments and plan IT audits based on control priorities.
3. Evaluate the effectiveness of information systems controls and configurations.
4. Analyze compliance against regulatory standards like ISO 27001 or GDPR.
5. Prepare professional audit reports with prioritized recommendations.

UNIT I: Foundations of IT Audit and Risk-Based Planning (12 Hours)

Introduction to Auditing in IT Environments, Types of Audits: Compliance, Operational, Technical, Financial, Audit Lifecycle: Planning, Fieldwork, Reporting, Follow-Up, Risk-Based Audit Approach, Internal Controls: Preventive, Detective, Corrective, Understanding Control Objectives, Risk Appetite and Materiality, COSO Framework

Outcome: Students understand why, when, and how audits are planned and scoped.

UNIT II: Audit of IT Infrastructure and Applications (12 Hours)

Network Audit Basics: Firewall, IDS/IPS, Router Configs, Application Controls Audit: Input, Processing, Output, Logical Access Control Reviews, Change Management and Patch Audits, Cloud & SaaS Audit Considerations, Business Continuity and Disaster Recovery Audit, Case Study: Active Directory Audit, ERP/CRM Security Evaluation

Outcome: Learners develop the ability to verify system configurations and security controls.

UNIT III: Governance, Compliance and Audit Reporting (12 Hours)

ISMS and ISO/IEC 27001 Auditing, COBIT 5 and 2019 Process Domains, GDPR and Data Audit Essentials, PCI-DSS and Financial System Audits, Audit Documentation and Working Papers, Evidence Collection and Sampling, Reporting Audit Findings: Severity, Risk, Remediation, Communicating with Management and Board

Outcome: Students connect standards to audit outcomes and learn to communicate insights.

Textbooks & Resources

Primary Textbooks

1. **Sandra Senft, Frederick Gallegos** – ***Information Technology Control and Audit***
   (CRC Press)
   ➤ A gold-standard academic text covering IT audit holistically.
2. **ISACA** – ***CISA Review Manual***
   ➤ For certification-level audit clarity and framework alignment.

Reference Books

- **David Coderre** – ***IT Audit: Control, Security, and Assurance***
- **COBIT Framework – ISACA Official Guide**
- **ISO 27001 Audit Guide (BSI / DNV)**
- **PCI-DSS v4.0 Implementation Guide**

Supplementary Platforms & Tools

- https://www.isaca.org – CISA, COBIT resources
- https://www.iso.org – ISO standards
- **Sample Audit Templates**, **Checklists**, **Interview Logs**
- Audit tools: **Nessus**, **Open-AudIT**, **Qualys**, **Splunk (for Log Audit)**
- Case studies: Equifax, Target breach, Aadhaar audit reports (public domain)

Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1: Describe audit frameworks and lifecycle in IT and cybersecurity domains	√						√			√
CO2: Conduct risk assessments and plan IT audits based on control priorities	√			√			√			
CO3: Evaluate the effectiveness of information systems controls and configurations	√	√		√	√					
CO4: Analyze compliance against regulatory standards like ISO 27001 or GDPR	√			√			√			
CO5: Prepare professional audit reports with prioritized recommendations				√				√	√	

SEMESTER – IV	
CSCP 560B	**INTERNSHIP + DISSERTATION6 (IN-CAMPUS)**

Course Outcomes

By the end of this course, the student should be able to:

- Independently identify, frame, and solve cybersecurity challenges.
- Use real tools and platforms as a **practitioner**, not just a student.
- Communicate findings effectively to both technical and managerial audiences.
- Demonstrate professionalism, precision, and ethical sensitivity.
- Build a **portfolio-ready artifact**—a system, a report, a tool, or a simulation.

Vision & Intentions

1. **Bridge Theory to Practice**: Apply academic learning to real-world cybersecurity problems, tools, and frameworks.
2. **Craft the Cyber Professional**: Develop a mindset of responsibility, precision, and ethical decision-making in cyber operations.
3. **Encourage Depth**: Go beyond project-based thinking; engage in **systematic investigation, modeling, or threat resolution**.
4. **Build Career Competence**: Offer a sandbox to simulate or contribute to SOCs, Red/Blue teaming, audits, secure app development, GRC, or digital forensics.
5. **Celebrate Individuality**: Each student builds a distinct practitioner's identity—researcher, defender, analyzer, ethical hacker, educator, innovator.

Guidelines & Modalities

🔹 Project Formats (Choose one)

- **Applied Project**: Set up simulations or real environments to test a threat model, analyze attacks, or build secure systems.
- **Research Dissertation**: Explore a novel idea in cyber policy, threat intelligence, AI for security, blockchain, etc.
- **Development-Based Work**: Create or enhance a tool, library, scanner, or automation script (e.g., custom log parser, IDS ruleset).
- **Audit/Review**: Perform structured security audit of a system/network using standard frameworks (e.g., ISO 27001, OWASP).
- **CTF-Driven Artifact**: Document lessons from Capture The Flag challenges, reverse engineering, or vulnerability analysis.

Supervision & Evaluation

- Assigned Faculty Supervisor + Industry Mentor (if applicable)

- Mid-Semester Review (Week 8–9)
- Final Viva with External Examiner Panel (Week 16–18)

Deliverables

1. **Project Proposal** (2–3 pages): Problem, scope, tools, methodology
2. **Work Log / Weekly Diary** (signed weekly)
3. **Final Report** (60–80 pages with screenshots, citations, logs, annexures)
4. **Live Demo or Simulation** (if applicable)
5. **Poster or Slide Deck** for defense
6. **Self-Reflection Note**: "How this project shaped me as a cybersecurity practitioner"

Suggested Areas of Work

- Threat Hunting and Log Analysis
- SIEM Rule Tuning (e.g., with Splunk, Wazuh)
- Malware Reverse Engineering
- Network Forensics Case Simulation
- Secure Code Review + OWASP Remediation
- Smart Contract Security Auditing
- Cyber Risk and GRC Dashboard
- Phishing Simulation Campaign & Analysis
- Cryptography API Implementation or Break
- SOC Monitoring Simulation / Runbook Drafting
- AI/ML Model for IDS or Email Filtering
- *Privacy-preserving Data Sharing Models*

Course Outcome	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10
CO1: Independently identify, frame, and solve cybersecurity challenges	✓	✓		✓	✓	✓				✓
CO2: Use real tools and platforms as a practitioner		✓		✓	✓	✓			✓	
CO3: Communicate findings effectively								✓		
CO4: Demonstrate professionalism and ethics				✓			✓	✓		
CO5: Build a portfolio-ready artifact		✓		✓	✓					✓