Session 10A

SSL and TLS

Mouli Sankaran

Session 10A: Focus

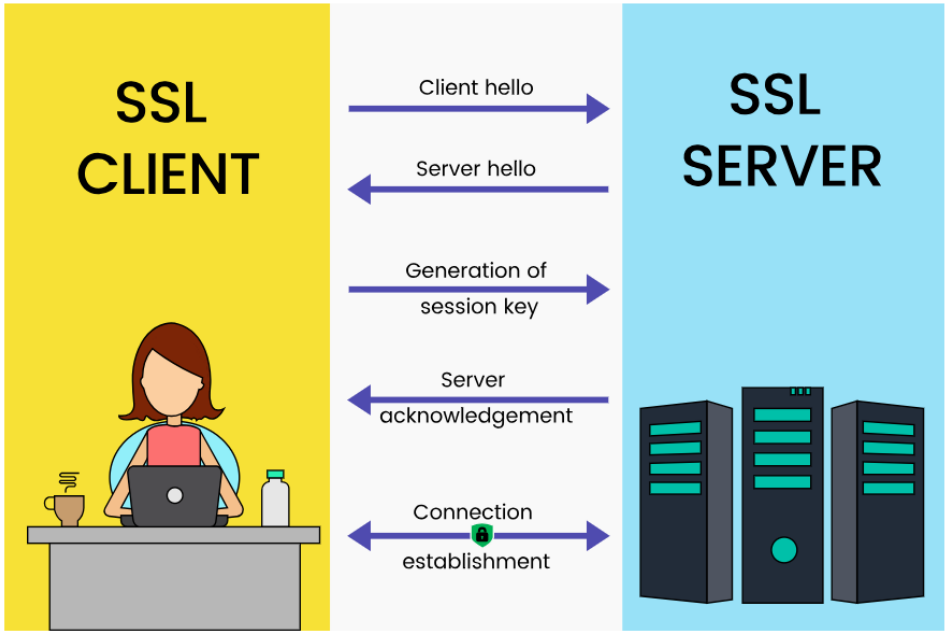
- SSL/TLS
  - Handshake
  - Applications
- DNS Security
  - Requirements
  - Key Features
- Cloud Security
  - Requirements and Key features
  - Challenges

Course page where the course materials will be posted as the course progresses:

Introduction to SSL/TLS

SSL/TLS: Introduction

- **SSL** (Secure Sockets Layer) is a **cryptographic protocol** designed to provide secure communication over a computer network.
- Developed originally by Netscape in the mid-1990s.



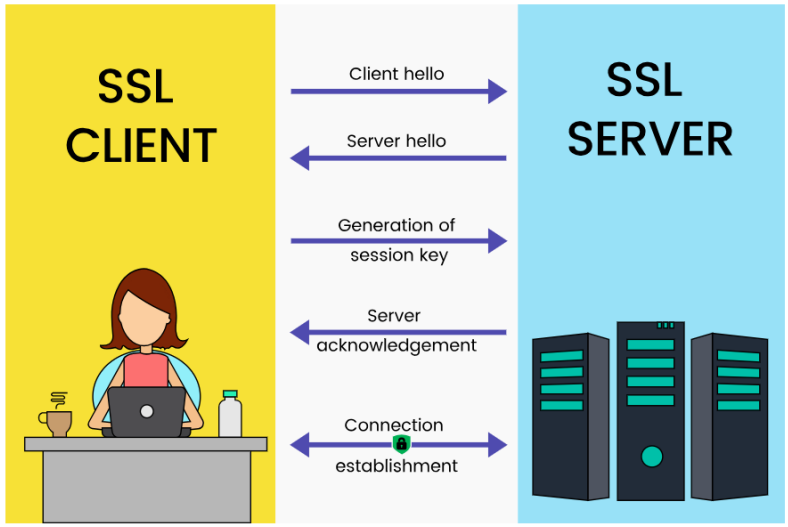
- It ensures that data sent between a client (like a web browser) and a server (like a website) is encrypted, authenticated, and protected against tampering.
- SSL has now evolved into **TLS** (Transport Layer Security) — TLS 1.0 was based on SSL 3.0 — but people still often use "SSL" informally to refer to both.

SSL/TLS: Salient Features

Feature	Explanation
Confidentiality	SSL encrypts the data so that it cannot be read by anyone except the intended recipient.
Integrity	SSL ensures that data has not been altered in transit using MACs (Message Authentication Codes).
Authentication	SSL allows the client to verify the server's identity using **digital certificates** (server authentication).
Optional Client Authentication	SSL can also optionally authenticate clients, though usually only servers are authenticated.
Session Keys	SSL uses **public key cryptography** to exchange a **symmetric session key** for faster encrypted communication.
Negotiation of Cipher Suites	SSL allows the client and server to agree on which encryption algorithms to use.
Forward Secrecy (optional)	Modern implementations (TLS 1.2+) can use ephemeral keys to ensure session keys are not compromised even if the private key is.

SSL: Handshake

- **Client Hello**: Client sends supported SSL/TLS versions, list of supported cipher suites, and a random number.
- **Server Hello**: Server responds with selected cipher suite, its digital certificate (public key), and its own random number.



- **Server Certificate Verification**: Client verifies the server's digital certificate using trusted Certificate Authorities (CAs).
- **Key Exchange**: Client generates a pre-master secret, encrypts it with the server's public key, and sends it to the server.
- Both client and server derive the same session keys from the pre-master secret and random numbers.
- **Finished Messages**: Both sides confirm that future communication will be encrypted using the newly established session key.

SSL/TLS: Data Communication

- The session key (symmetric key) is now used for encrypting all the subsequent data between client and server.
- Symmetric encryption is faster than public-key encryption, hence session keys are preferred for ongoing data transfer.

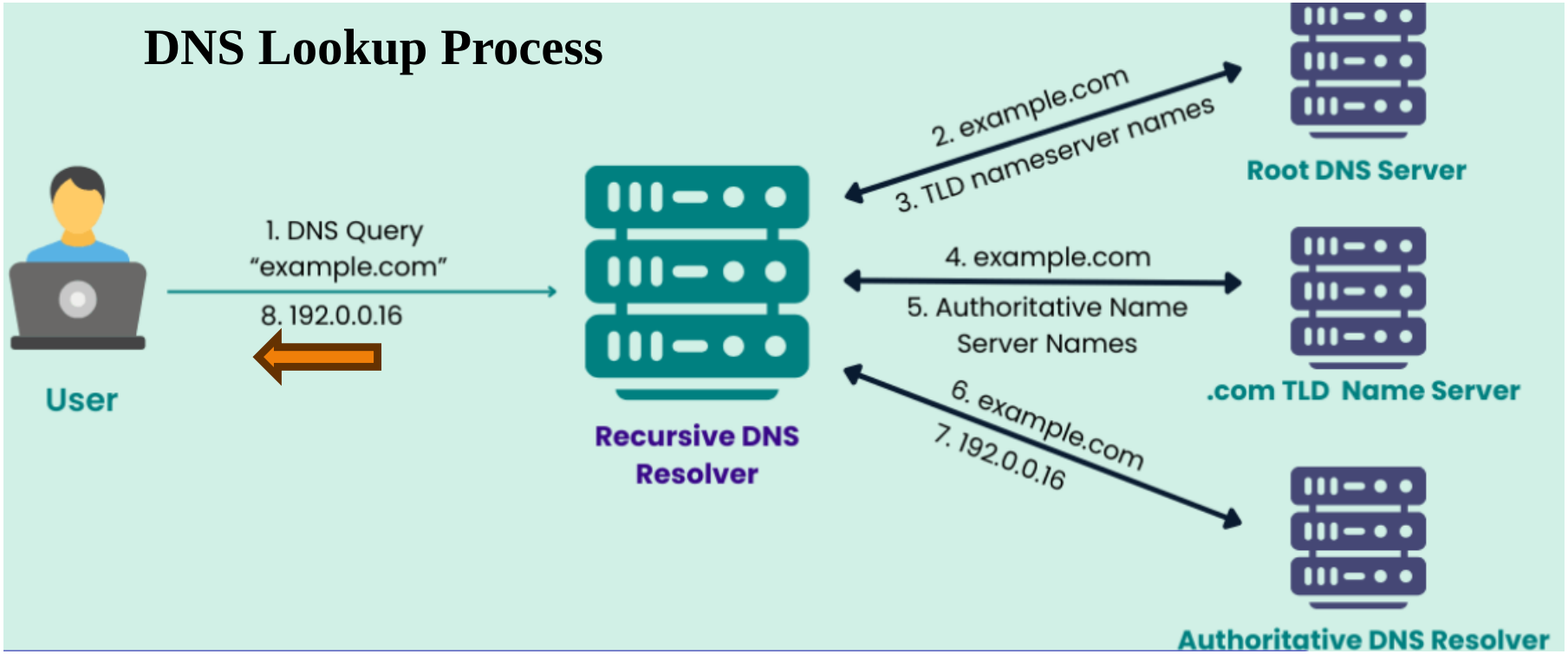
SSL/TLS: Applications

Application	Use of SSL
Web Browsing (HTTPS)	SSL secures HTTP connections to websites.
Email	SSL/TLS used to secure SMTP (sending), IMAP/POP3 (retrieving emails).
VPNs	SSL VPNs secure remote access over public networks.
Online Banking and Payments	SSL protects sensitive financial data.
Instant Messaging and VOIP	SSL can encrypt chat messages and voice calls.
E-commerce Transactions	SSL protects credit card information during online purchases.
Software Updates	SSL secures the download of software patches to prevent tampering.

DNS Introduction

DNS: Domain Name Server

- DNS is the behind-the-scenes service that makes it possible to browse the web by converting human-friendly domain names into IP addresses that computers can understand.



DNS Query Resolution

- **Client Interaction**: When a domain name (e.g. www.example.com) is entered into a browser, the request is first sent to the **recursive resolver**.
- This is typically provided by the Internet Service Provider (ISP) or a public DNS resolver like Google's Public DNS (8.8.8.8).
- **Recursive Queries**: If the resolver doesn't have the answer to the query stored, it performs recursive queries, contacting other DNS servers in the hierarchy (Root, TLD, and Authoritative) on behalf of the client to find the correct IP address for the domain.
- **DNS Caching**: To improve efficiency and reduce lookup times, the recursive resolver stores the results of DNS queries in a cache for a period of time.
- This allows future requests for the same domain to be resolved more quickly.

DNS Security

DNS Security

- The original DNS was not designed with security in mind.
- It trusts that all responses it receives are legitimate.
- This makes it vulnerable to attacks such as:
- **DNS Spoofing** (forging responses)
- **Cache Poisoning** (injecting false entries into a resolver's cache)
- **Man-in-the-Middle (MITM) attacks**
- Thus, securing DNS is vital for maintaining the integrity and trustworthiness of Internet communications

DNS Security: Requirements

Requirement	Description
Authentication of DNS Responses	Clients must be able to verify that the response they receive matches the response intended by the domain owner.
Protection of Zone Data	DNS zones must be securely managed and cryptographically signed to prevent tampering.
Key Management	Proper handling of cryptographic keys (generation, storage, rollover, revocation) is necessary to maintain security.
Chain of Trust	Validation must extend up the DNS hierarchy — from root, to top-level domain (TLD), to domain.
Resilience to Replay Attacks	Signed responses must include mechanisms like nonces or timestamps to prevent attackers from replaying old responses.
Availability and Performance	Security mechanisms should not significantly degrade the performance of DNS resolution.
Operational Simplicity	The system must allow for relatively easy deployment and management by operators (e.g., auto-signing zones).

DNS Security: Key Features

Feature	Description
Data Integrity	Ensures that the DNS responses have not been tampered with.
Authentication of Source	Verifies that DNS data comes from the authoritative source (not a fake server).
Protection Against Spoofing and Poisoning	Prevents attackers from injecting fake DNS data into a resolver's cache.
Cryptographic Assurance	Uses digital signatures to validate DNS information.
Backward Compatibility	Secure extensions (like DNSSEC) work alongside traditional DNS systems where possible.
Minimal Confidentiality	DNS traditionally does not encrypt queries; however, newer extensions like **DoT** (DNS over TLS) and **DoH** (DNS over HTTPS) provide confidentiality for DNS queries.

DNS Security: Benefits

- **Prevents redirection to malicious websites** (e.g., phishing, malware sites).
- **Strengthens Internet trustworthiness** by ensuring domain-to-IP mappings are correct.
- **Hardens critical infrastructure** (especially important for enterprises, ISPs, and governments).
- **Supports secure email delivery** (DNSSEC supports secure email via technologies like DANE).

DANE (DNS-based Authentication of Named Entities) is a DNS security protocol that binds digital certificates to domain names using DNSSEC

Cloud Security

Cloud Security: Introduction

- **Cloud computing** allows on-demand access to computing resources like servers, storage, and applications over the Internet.
- However, moving to the cloud introduces new security challenges — like shared environments, multi-tenancy, remote access, and outsourced management.
- **Cloud Security** refers to the technologies, policies, controls, and procedures used to protect cloud-based systems, data, and infrastructure.

Cloud Security: Requirements

Requirement	Description
Data Confidentiality, Integrity, and Availability (CIA Triad)	Protect cloud data from unauthorized access, tampering, and ensure it is available when needed.
Secure Access Controls	Implement strong authentication (MFA), role-based access control (RBAC), and least privilege principles.
Encryption Mechanisms	Use strong encryption for data at rest, in transit, and optionally during processing (homomorphic encryption).
Shared Responsibility Model	Understand that security responsibilities are split between the **cloud provider** and the **cloud customer**.
Security Configuration Management	Secure setup of virtual machines, storage, databases, containers, and networking components.
Auditability and Logging	Maintain logs of user activities, system changes, and security events for accountability and investigations.
Vulnerability Management	Regularly scan and patch systems to fix known security vulnerabilities.

Cloud Security: Key Features

Feature	Explanation
Data Protection	Data at rest, in transit, and in use must be encrypted and securely handled.
Access Control and Identity Management	Only authorized users and systems should have access to cloud resources (IAM, MFA).
Network Security	Protect cloud networks against intrusion, DDoS attacks, and unauthorized access.
Visibility and Monitoring	Continuous monitoring of cloud resources for abnormal activity or breaches.
Compliance and Legal Protection	Support for regulatory compliance (e.g., GDPR, HIPAA, PCI-DSS).
Incident Response	Ability to detect, respond to, and recover from security incidents quickly.
Isolation and Multi-Tenancy Control	Prevent data leaks and interference between different tenants (customers) sharing the same cloud.
Availability and Disaster Recovery	Ensure cloud services are resilient against failures and can recover rapidly.
Security Automation	Automated threat detection, policy enforcement, and incident handling to deal with cloud scale and speed.

Cloud Security: Challenges

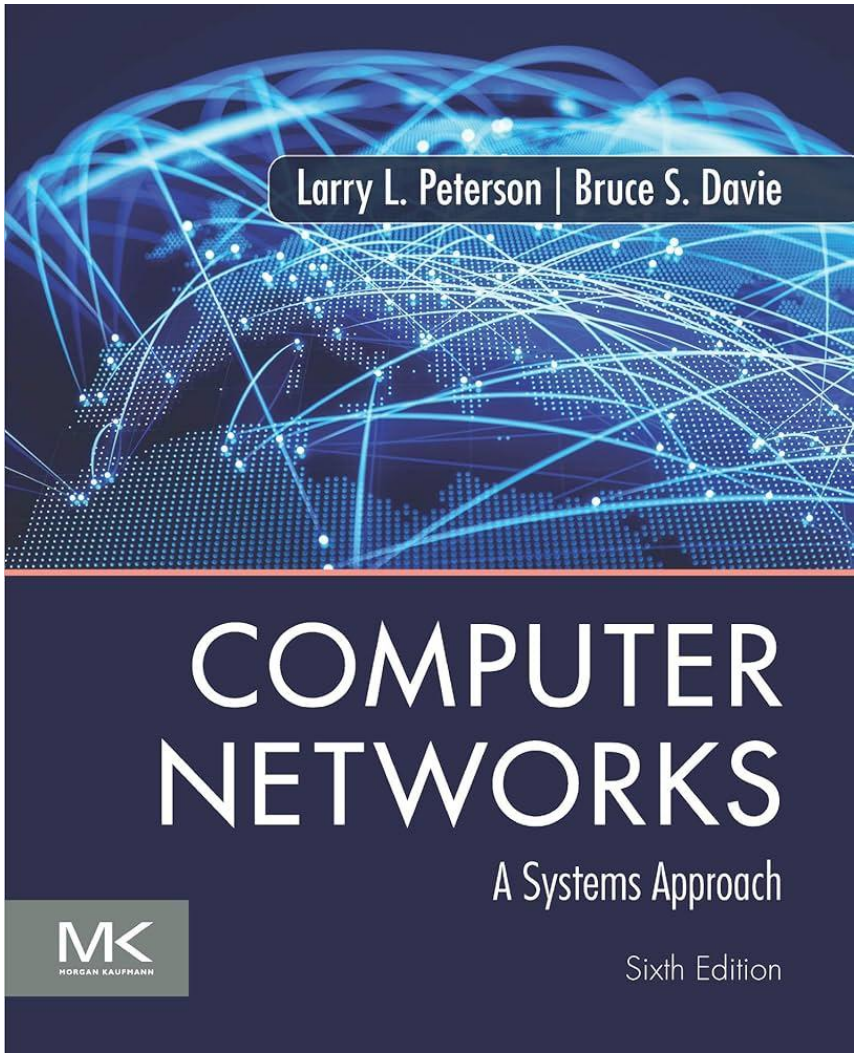
Challenge	Explanation
Loss of Control	Data and infrastructure are managed by a third party.
Shared Technology Vulnerabilities	Multi-tenancy increases risk if virtualization/isolation fails.
Data Breaches and Misconfigurations	One of the top causes of cloud breaches (e.g., misconfigured S3 buckets).
Insider Threats	Threats from employees or administrators inside the cloud organization.
Compliance Complexity	Different countries have different data privacy and security laws.

Session 10A: Summary

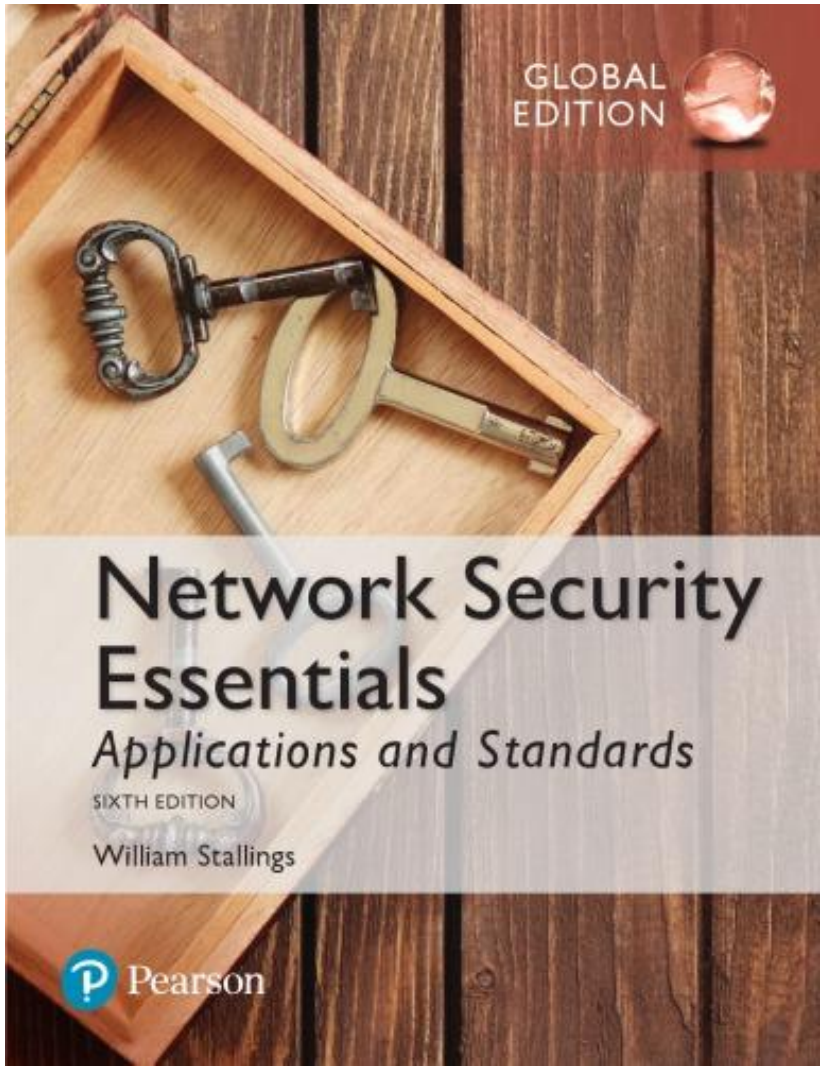
- SSL/TLS
  - Handshake
  - Applications
- DNS Security
  - Requirements
  - Key Features
- Cloud Security
  - Requirements and Key features
  - Challenges

Textbooks

Textbook 1

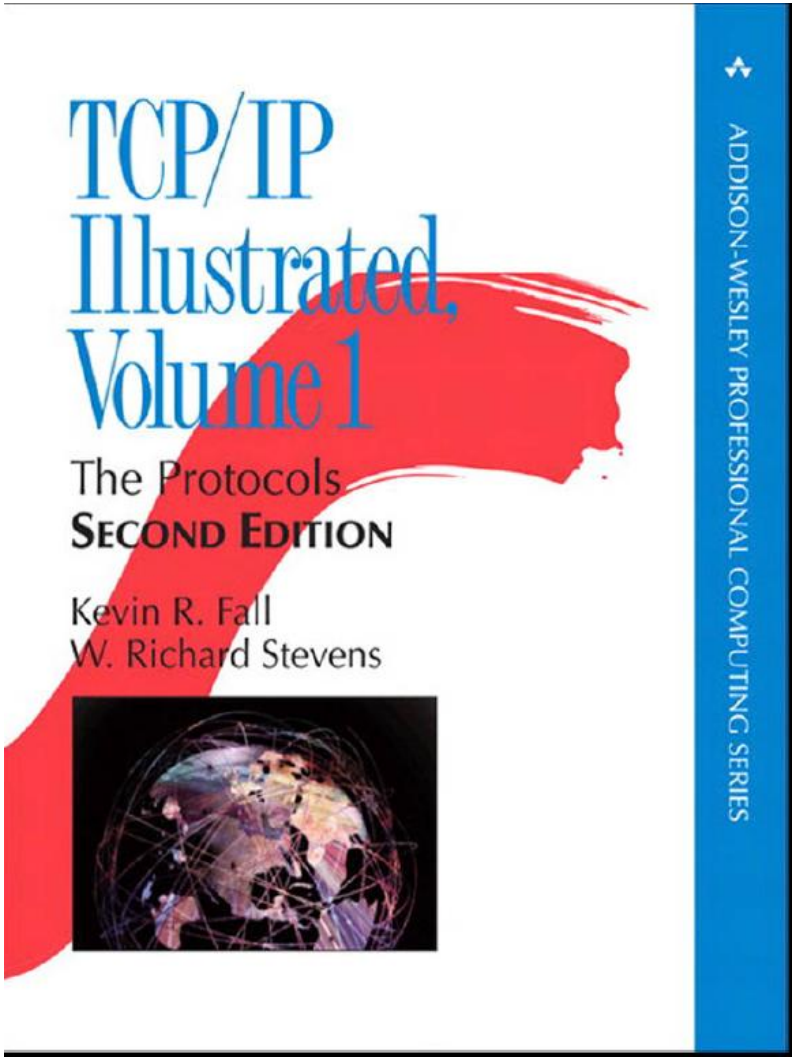


Textbook 2



References

Ref 1



Ref 2

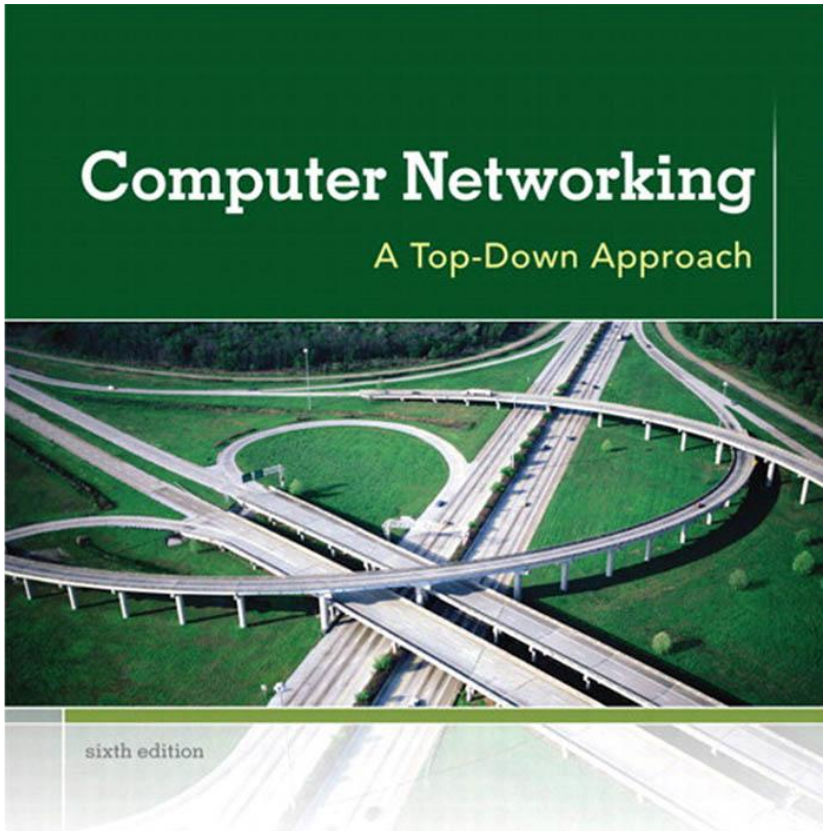
TCP Congestion Control: A Systems Approach

TCP Congestion Control: A Systems Approach

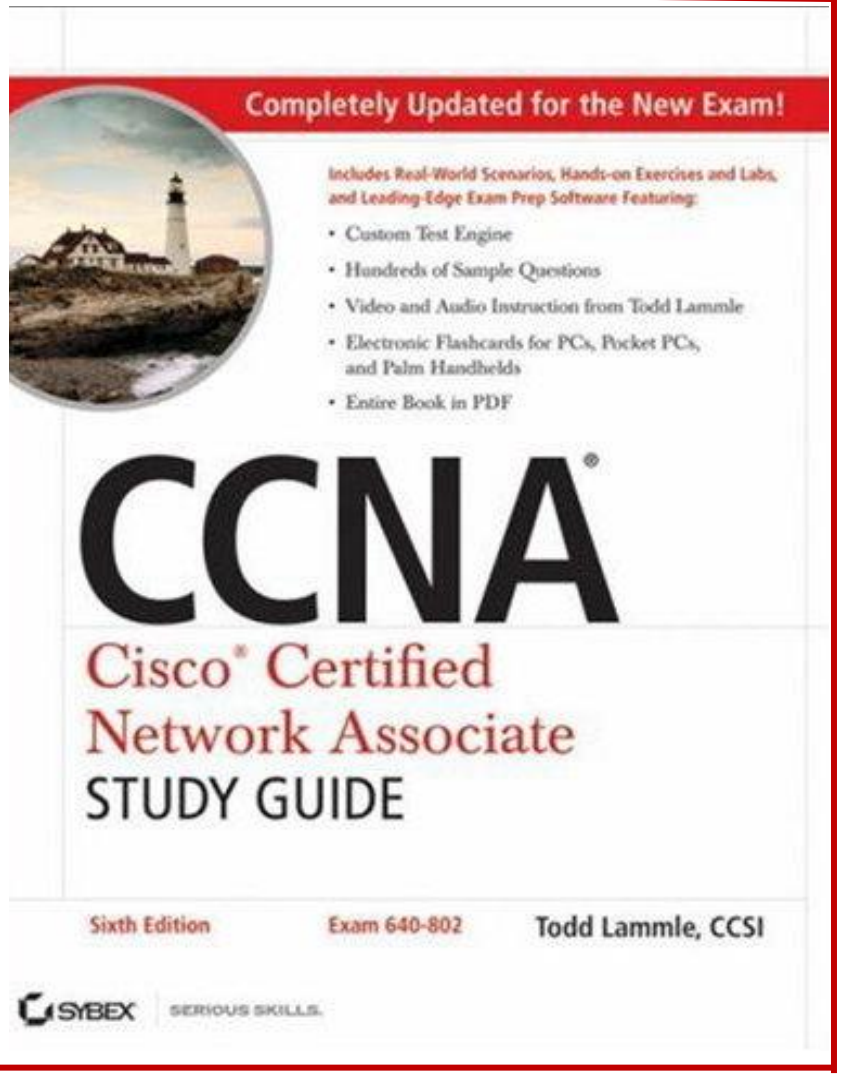
Peterson, Brakmo, and Davie

References

Ref 3



Ref 4



References

Ref 5