Essential Ports for DevOps Engineers

1. **HTTP (Hypertext Transfer Protocol) - Port 80**

   Purpose: **Used for transferring web pages.**

   - **Key Points**:
     - Facilitates communication between web browsers and servers.
     - Stateless protocol; each request is independent.
     - Commonly used for unencrypted web traffic.

2. **HTTPS (HTTP Secure) - Port 443**

   Purpose: **Secure version of HTTP.**

   - **Key Points**:
     - Encrypts data using SSL/TLS for secure communication.
     - Essential for handling sensitive data online.
     - Widely adopted as the standard for all websites.

3. **SSH (Secure Shell) - Port 22**

   Purpose: Secure remote login and command execution.

   - **Key Points**:
     - Provides a secure channel over unsecured networks.
     - Supports encrypted file transfers (SCP, SFTP).
     - Commonly used in server management.

4. **FTP (File Transfer Protocol) - Port 21**

   Purpose: Transfers files between computers.

   - **Key Points**:
     - Allows users to upload/download files from servers.
     - Can be insecure; often replaced by SFTP or FTPS.

- Supports both anonymous and authenticated access.

5. **SFTP (SSH File Transfer Protocol) - Port 22**

   Purpose: Secure file transfer protocol.

   - **Key Points**:
     - Uses SSH to provide secure file transfer capabilities.
     - Combines file access, transfer, and management functionalities.
     - More secure than traditional FTP.

6. **SMTP (Simple Mail Transfer Protocol) - Port 25**

   Purpose: Sending emails across networks.

   - **Key Points**:
     - Essential for email delivery from client to server.
     - Often used with POP3 or IMAP for retrieving emails.
     - Vulnerable to spam; many ISPs restrict its use.

7. **DNS (Domain Name System) - Port 53**

   Purpose: Resolves domain names into IP addresses.

   - **Key Points**:
     - Critical for routing internet traffic correctly.
     - Operates using both TCP and UDP protocols.
     - DNS caching improves performance and reduces server load.

8. **MySQL Database Server – Port 3306**

   Purpose: Default port for MySQL connections.

   - **Key Points**:
     - Used by applications to connect securely to MySQL databases.
     - Supports various authentication methods for security.
     - Often configured with firewalls to restrict access.

9. **PostgreSQL Database Server – Port 5432**

   Purpose: Default port for PostgreSQL database connections.

   - **Key Points**:
     - Supports advanced data types and performance optimization features.
     - Uses SSL for secure connections.

- Commonly used in web applications and enterprise systems.

10. **Docker Daemon API – Port 2375/2376**

Purpose: Ports used by the Docker daemon for API access.

- **Key Points**:
  - Port 2375 is unsecured; port 2376 uses TLS encryption.
  - Essential for managing Docker containers remotely.
  - Requires careful configuration to avoid security vulnerabilities.

11. **Kubernetes API Server – Port 6443**

Purpose: Entry point for all API requests in Kubernetes clusters.

- **Key Points**:
  - Handles REST requests and updates the cluster state accordingly.
  - Secured with TLS to ensure communication integrity and confidentiality.
  - Essential for managing resources within Kubernetes.

12. **Redis Server – Port 6379**

Purpose: Default port used by Redis, an in-memory data structure store.

- **Key Points**:
  - Used as a database, cache, and message broker.
  - Supports various data structures such as strings, hashes, lists, etc.
  - Often used in microservices architectures for fast data access.

13. **MongoDB – Port 27017**

Purpose: Default port for MongoDB database connections.

- **Key Points**:
  - NoSQL database designed for scalability and performance.
  - Uses BSON format to store data, facilitating complex queries.
  - Supports horizontal scaling through sharding.

14. **Apache Kafka – Port 9092**

Purpose: Default port for Apache Kafka messaging system connections.

- **Key Points**:
  - Used for building real-time data pipelines and streaming applications.
  - Highly scalable and fault-tolerant architecture.
  - Supports pub/sub messaging model.

15. **Nginx – Port 80/443**

Purpose: Web server that can also act as a reverse proxy server, load balancer, etc.

- **Key Points**:
  - Handles static content efficiently; commonly used with dynamic content servers like Node.js or PHP-FPM.
  - Can serve as a load balancer to distribute traffic across multiple servers.

16. **Tomcat – Port 8080**

Purpose: Default port for Apache Tomcat web server instances serving Java applications.

- **Key Points**:
  - Implements Java Servlet and JavaServer Pages technologies.
  - Commonly used in enterprise environments to run Java-based web applications.

17. **VNC (Virtual Network Computing) – Port 5900**

Purpose: Remote desktop sharing system that uses the RFB protocol to remotely control another computer's desktop interface.

- **Key Points**:
  - Enables remote access to graphical desktop environments over a network connection.

18. **ElasticSearch – Port 9200**

Purpose: Default port used by ElasticSearch REST API calls.

- **Key Points**:
  - Distributed search engine built on Apache Lucene; used for full-text search capabilities and analytics on large datasets.

19. **RabbitMQ – Port 5672**

Purpose: Default port used by RabbitMQ messaging broker connections.

- **Key Points**:
  - Implements Advanced Message Queuing Protocol (AMQP).
  - Used in microservices architectures to facilitate communication between services.

20. **Grafana – Port 3000**

Purpose: Default port used by Grafana dashboard application for visualizing metrics from various data sources like Prometheus or InfluxDB.

- **Key Points**:
  - Provides powerful visualization capabilities and alerting features based on metrics collected from various systems.