**LAB 10**

**4.1.**

A) On which port number was the ftp data connection established?

1024

B) What is the essence of the passive data connection establishment?

**Passive Mode**:

* + In passive mode, the client uses a PASV (Passive) command.
  + The server responds with an IP address and a **random port number** (usually in the range 1024-65535).
  + The client then establishes a data connection by connecting to the server’s IP address on the specified port.
  + The essence of passive mode is that the **server provides the necessary information**, and the client connects to it.

**4.2.**

A) What is the port number of the DNS query?

**UDP 53**

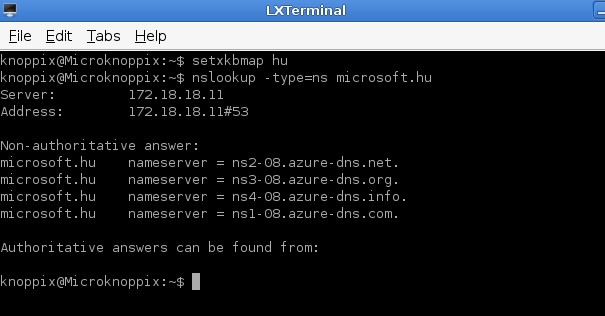
B) Was there any iterative queries? What can be the explanation for it?

* **Iterative Query**:
  + In an iterative query, the client asks the DNS server for resolution but expects a partial answer.
  + The DNS server provides the best information it has (often referring the client to other DNS servers).
  + The client then continues the resolution process by querying other DNS servers based on the information received.
  + This process repeats iteratively until the client obtains the final IP address.
* **Explanation for Iterative Queries**:
  + Iterative queries are essential for DNS scalability and efficiency.
  + They distribute the workload across multiple DNS servers.
  + If a DNS server doesn’t know the answer, it can refer the client to another server, reducing the load on authoritative servers.
  + Overall, iterative queries allow the DNS system to handle a large number of requests efficiently.

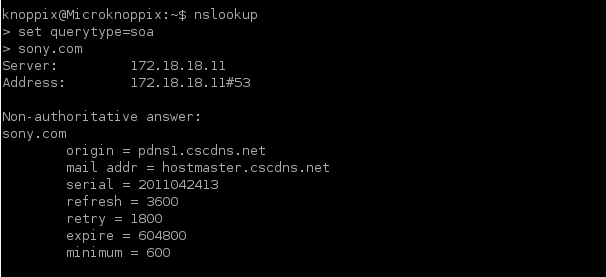
**4.3.**

4.3. A) How many secondary DNS servers does the microsoft.com domain have?

It can vary.



B) What period of time is the primary DNS server queried by the secondary one in the case of sony.com domain?



**4.4.**

A) How can we make sure that we accessed the data from the web server without a problem?

1. **HTTP Status Codes**:
   * When you access a web server, it responds with an HTTP status code. These codes indicate whether the request was successful or encountered an issue.
   * Common status codes include:
     + **200 OK**: The request was successful, and the data is being served.
     + **404 Not Found**: The requested resource (e.g., page or file) doesn’t exist on the server.
     + **500 Internal Server Error**: The server encountered an error while processing the request.
     + **301/302 Redirect**: The server redirects the request to a different URL.
   * Monitor the status codes in the server response to ensure everything is functioning as expected.
2. **Network Connectivity and Latency**:
   * Check your network connection. Ensure you can reach the web server’s IP address or domain.
   * High latency can impact data retrieval. Use tools like **ping** or **traceroute** to diagnose network issues.
3. **DNS Resolution**:
   * The Domain Name System (DNS) translates domain names (e.g., www.example.com) into IP addresses.
   * Verify that DNS resolution is working correctly. Use the **nslookup** or **dig** command to check DNS records.
4. **Firewalls and Security Settings**:
   * Firewalls can block access to web servers. Ensure that your firewall rules allow traffic to the server’s port (usually port 80 for HTTP).
   * Check security settings (e.g., SSL/TLS certificates) to ensure secure communication.
5. **Content Integrity**:
   * Verify that the data received matches what you expect. Check for any unexpected modifications or corruption.
   * Use checksums or cryptographic hashes to validate data integrity.
6. **Server Logs and Monitoring**:
   * Web servers maintain logs. Review these logs for any errors, warnings, or unusual activity.
   * Set up monitoring tools to track server performance, resource usage, and potential issues.

B) What does the HTTP GET packet contain?

1. **Request Line**:
   * The first line of the GET packet contains the request method, the requested resource (usually a URL), and the HTTP version.
   * Example: GET /index.html HTTP/1.1
2. **Headers**:
   * Headers provide additional information about the request. Some common headers include:
     + **Host**: Specifies the domain name of the server.
     + **User-Agent**: Identifies the client (e.g., browser or application) making the request.
     + **Accept**: Indicates the preferred content type (e.g., HTML, JSON, XML).
     + **Connection**: Specifies whether the connection should be kept alive.
   * Example:
   * Host: www.example.com
   * User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.3
3. **Empty Line**:
   * An empty line separates the headers from the optional message body.
4. **Message Body (Optional)**:
   * The GET request typically doesn’t include a message body. However, some APIs or services may use it to send data (e.g., in a POST request).
   * Example (not common for GET):
   * Content-Type: application/json
   * Content-Length: 42
   * {"key": "value"}
5. **Example GET Request**:
6. GET /index.html HTTP/1.1
7. Host: www.example.com

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/58.0.3029.110 Safari/537.3