

1

ABOUT MICROSERVICES

ARCHITECTURE

- Multithreaded, Pooled, Containers, Nodes
 Load Balancer/API Gateway

vs MONOLITHIC

- Scalable, Robust, and Platform-Independent
- Language/Transport/Protocol Agnostic

FUNCTIONALITY

- Compute
- Use APIs
- Access Data (static files as well as databases)
- Delegate / Orchestrate with other services

2



ABOUT TCP

- TCP over IP
- IP: machine-to-machine (IP address)
- Loopback/Localhost; Private/Public IP; NAT
- TCP: process-to-process (port number)
- Connection: order + guaranteed delivery
- Protocol: text/binary; custom/standard
- Connection: yes; State: no (restful)

4



5



6

TCP CLIENT

- Find the server's Host and Port Socket client = new Socket(host, port);
- Get a stream to issue the request to it new PrintStream(client.getOutputStream(), true).println(request...);

See course repo

- Get a stream to read the response from response = new Scanner(client.getInputStream()).nextLine();
- Close the connection client.close();

7

THREADS IN JAVA

- Given a class W that extends Thread which means it has a public void run()
- To execute W "in parallel" with you new W().start() [this will invoke run()]
- In microservices, run() would typically: compute; use APIs; lookup a DB; use HTTP; ...
- Other languages See the repo for Golang and Python examples

8

TCP SERVER

- Create a server instance on a host/port ServerSocket server = new ServerSocket(port, 0, host);
- Query the server re host/port for logging server.getInetAddress(), server.getLocalPort()
- Socket socket = server.accept() accept will block until a request arrives
- Query the socket getPort, getInetAddress, getInput/OutputStream

EECS4413/Roumani



DATABASE ACCESS

- Determine the database URL jdbc:sqlite:<path to db file> for file-based / embedded jdbc:derby://host:port/name;credentials for network-based
- Get a connection to the DB Connection connection = DriverManager.getConnection(url...);
- Create a SQL statement
 Statement statement = connection.createStatement();
 PreparedStatement stm = con.prepareStatement(query);
- Execute the query and see its result *ResultSet rs = statement.executeQuery(query...)* Can then do if or while based on *rs.next()*

Eclipse has tools (Data Source Explorer, SQL Scrapbook) to explore the database and test queries. For SQLite, I prefer CLI –see https://sqlite.org/cli.html 11

11

XML MARSHALLERS

- JAXBContext context = JAXBContext.newInstance(bean⁺.class);
- Marshaller m = context.createMarshaller();
- m.setProperty(Marshaller.JAXB_FORMATTED_OUTPUT, Boolean.TRUE);
- m.marshal(<bean instance>, <output stream>);

The xml document root must be annotated: @XmlRootElement(name = "whatever")





14

ABOUT

- HTTP over TCP uses URI := URN | URL
- URL syntax: proto://host:port/path?QS
- QS: param1=val1¶m2=val2... The val are URL-encoded (%hex)
- HTTP Methods: GET, HEAD, POST
- Request / Response protocol
- The Response Status Codes 1xx, 2xx, 3xx, 4xx, 5xx
- Cousin Protocols HTTP 2.0, HTTPS, and WebSocket

15

HANDS-ON EXAMPL	E
GET /index.html HTTP/1.1 Host: roumani.eecs.yorku.ca	telhes
HTTP/1.1 200 OK Content-Type: text/html	to iti
<payload></payload>	
 GET Line, Host header line, optiona 	l header lines
 Empty line Status Line Content-type line ontic 	nal header lines
 Empty Line 	mar neader miles
 Pavload 	

16



17

SECURITY

- Network based Measures
 IP filtering, black and white lists, DOS, DDOS, ...
- Confidentiality Encrypt: link level or end-to-end
- Authentication Passwords, Accounts, OAuth
- SQL Injection Always sanitize incoming parameters and use prepared queries.

PASSWORDS 9. fne past: store password greer hacks; insider attacks; social engineering. 9. fne Present: store the hash dred equality; rainbow table > add salt and slow it down. 9. fne Future¹: No passwords! Why ot PKI with challenges? Password never leave you and cannot be re-played.

19

TELEMETRY

- System Logs
- Performance Timing
- Metrics
- Analytics
- Data Mining
- Machine Learning

20

SCALABILITY

- Multithreading
- Throttling
- Thread Pooling
- Scale thru Containers (e.g. Docker, Kubernetes)
- Scale thru boxes (pods and nodes) or VMs
- Load Balancing
- Auto Scaling