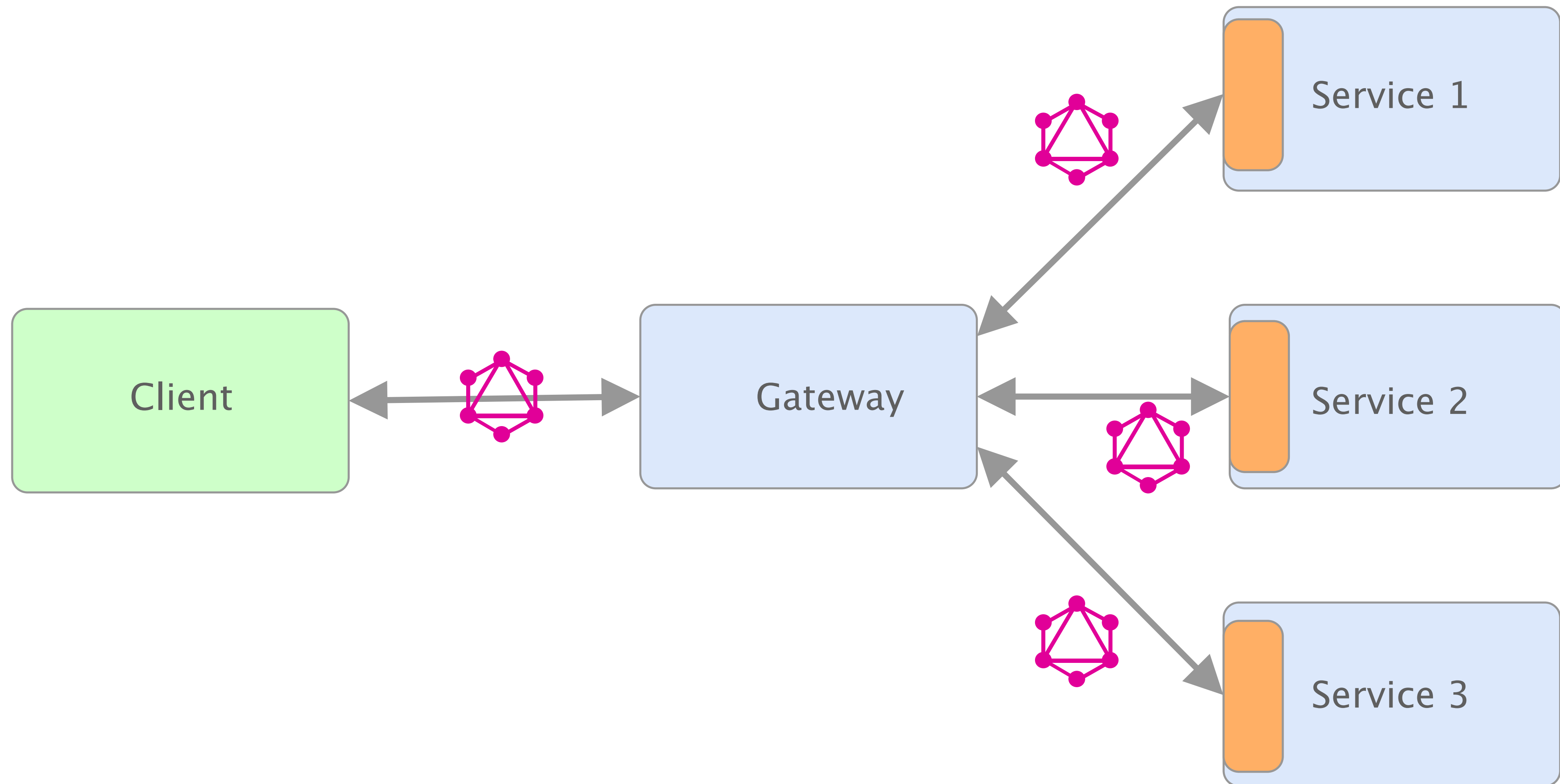
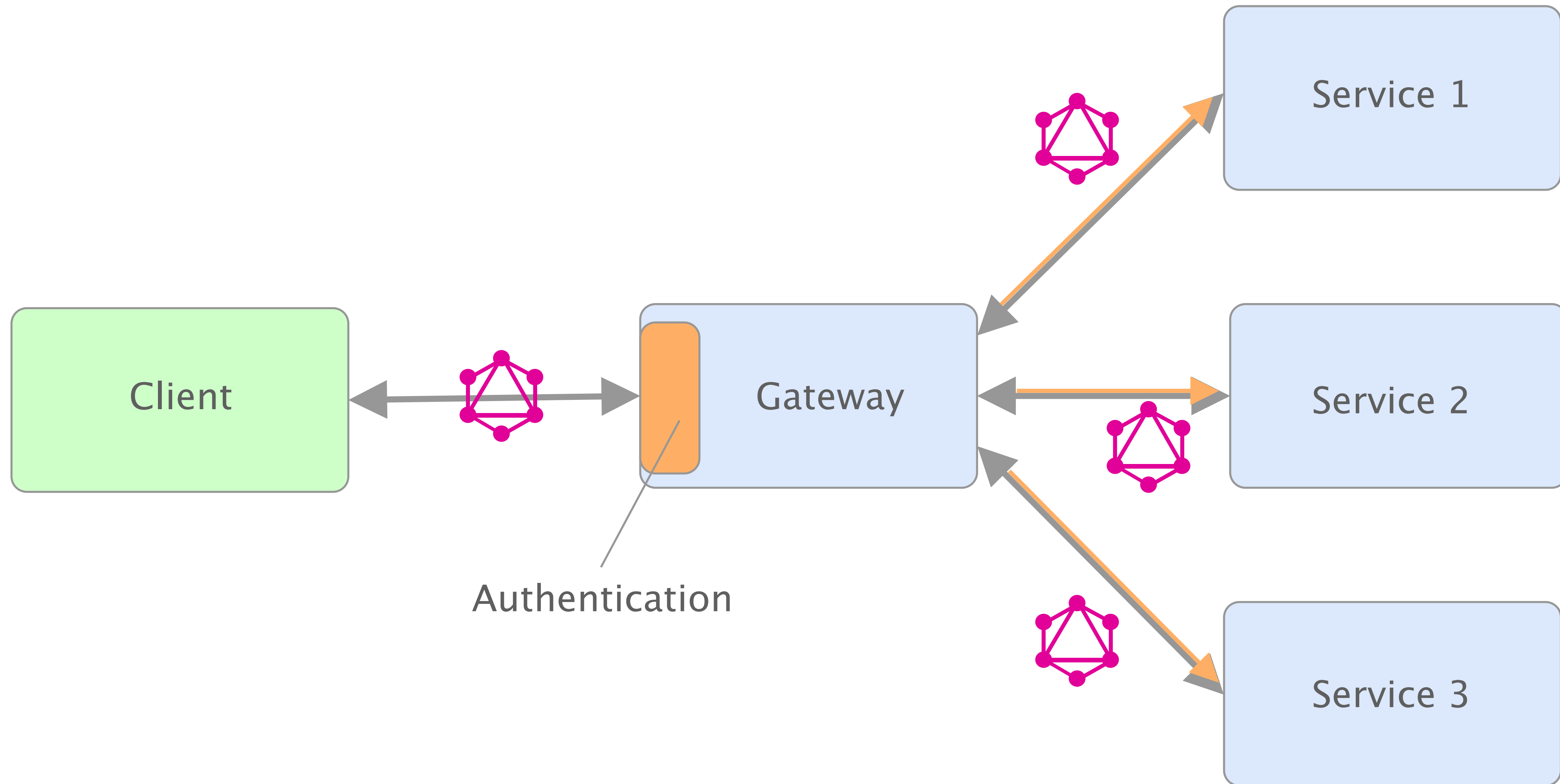


Authentication when Schema Stitching

Option 1: Pass token through



Option 2: Convert token to authenticated info



```
const { HttpLink } = require("apollo-link-http");
const { setContext } = require('apollo-link-context');
const fetch = require("node-fetch");

const schema = async () => {
  const httpLink = new HttpLink({ uri: "http://localhost:4000/", fetch });
  const link = setContext((_request, context) => ({
    headers: {
      'Authorization': `Bearer ${context.graphqlContext.authKey}`,
    }
  })).concat(httpLink);
  const schema = await introspectSchema(link);
  const executableSchema = makeRemoteExecutableSchema({
    schema,
    link
  });

  return executableSchema;
};

module.exports = schema
```

Exercise

Refactor the application in a way that token is converted to a user object inside the Gateway and pass the user along to the service using a header.

The services should be updated to leverage the user header.