JÖNKÖPING UNIVERSITY

School of Engineering

RUNNING NODE.JS ON LIGHTSAIL

Peter Larsson-Green

Lecturer at Jönköping University

Autumn 2021

WHAT IS AMAZON WEB SERVICES?

A collection of services from Amazon we can use:

- Elastic Compute Cloud (EC2): General (virtual) servers
- Relational Database Service (RDS): Servers for databases
- Simple Storage Service (S3): Storing files
- …

There are other alternatives:

- Amazon Web Services: https://aws.amazon.com/
- Google Cloud: https://cloud.google.com/
- Microsoft Azure: https://azure.microsoft.com/en-us/
- …

WHAT IS LIGHTSAIL?

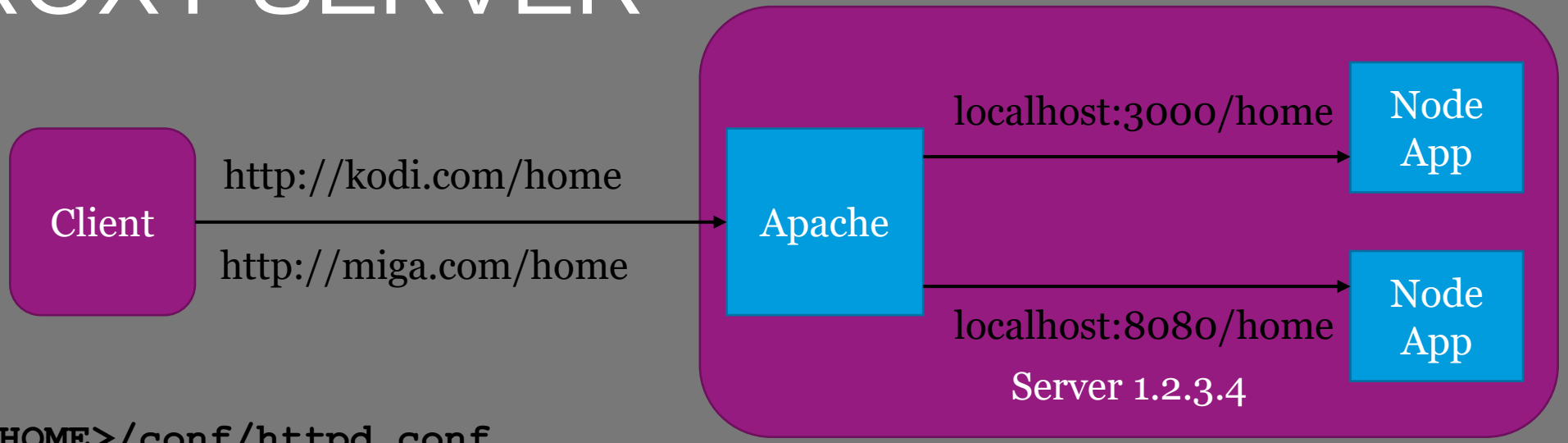
One of Amazon's services we can use to run web applications.

- https://aws.amazon.com/lightsail/

RUNNING ON PORT 8080

1. Upload your code
   - Have it in a Git repository? `git clone …`
   - Use an SFTP client: https://lightsail.aws.amazon.com/ls/docs/en_us/articles/amazon-lightsail-connecting-to-linux-unix-instance-using-sftp
     - FileZilla: https://filezilla-project.org/
     - …
2. `cd the-project-folder`
3. `npm install`
4. `node app.js`
   - `forever start app.js`
   - `forever list`
   - `forever stop processId`
5. Open port 8080 on the server instance

PROXY SERVER



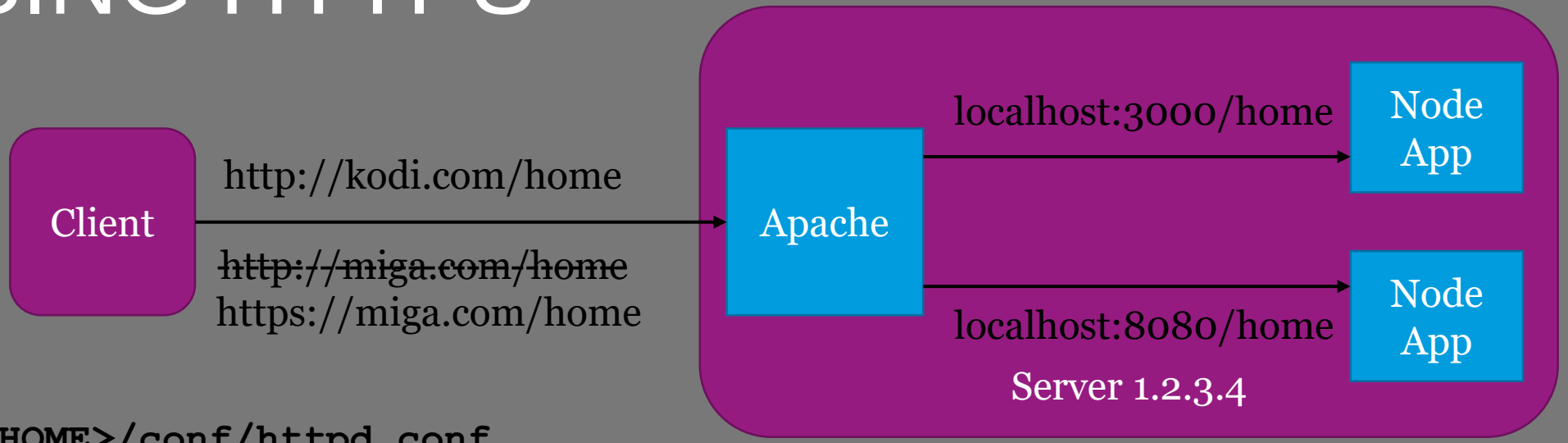
<APACHE_HOME>/conf/httpd.conf

```
Listen 80
<VirtualHost *:80>
    ServerName kodi.com
    ProxyPass / http://localhost:3000/
</VirtualHost>
<VirtualHost *:80>
    ServerName miga.com
    ProxyPass / http://localhost:8080/



</VirtualHost>
```

USING HTTPS



<APACHE_HOME>/conf/httpd.conf

```
Listen 443
Listen 80
<VirtualHost *:80>
    ServerName kodi.com
    ProxyPass / http://localhost:3000/
</VirtualHost>
<VirtualHost *:443>
    ServerName miga.com
    ProxyPass / http://localhost:8080/
    SSLEngine on
    SSLCertificateFile "/path/to/www.example.com.cert"
    SSLCertificateKeyFile "/path/to/www.example.com.key"
</VirtualHost>
```

HTTPS:

1. Get a domain name: https://internetstiftelsen.se/
2. Map the domain name to your server's IP address.
3. Create a certificate signed by a Certificate Authority: https://letsencrypt.org/getting-started/
4. Tell Apache about your certificate.

RUNNING WITH APACHE

One way to do it:

1. Start your app listening on port 8080.
2. `nano /opt/bitnami/apache/conf/vhosts/00_status-vhost.conf`

```
<VirtualHost *:80>
    ProxyPass / http://localhost:8080/
</VirtualHost>
```

3. `sudo /opt/bitnami/ctlscript.sh restart apache`