Securing Node.js

Agenda

- **Securing Node.js**
  - Express Pipeline
  - What is Passport?
  - Implementing Registration
  - Securing Passwords
  - Using Passport
  - Implementing Login
  - Authorizing Pages and API

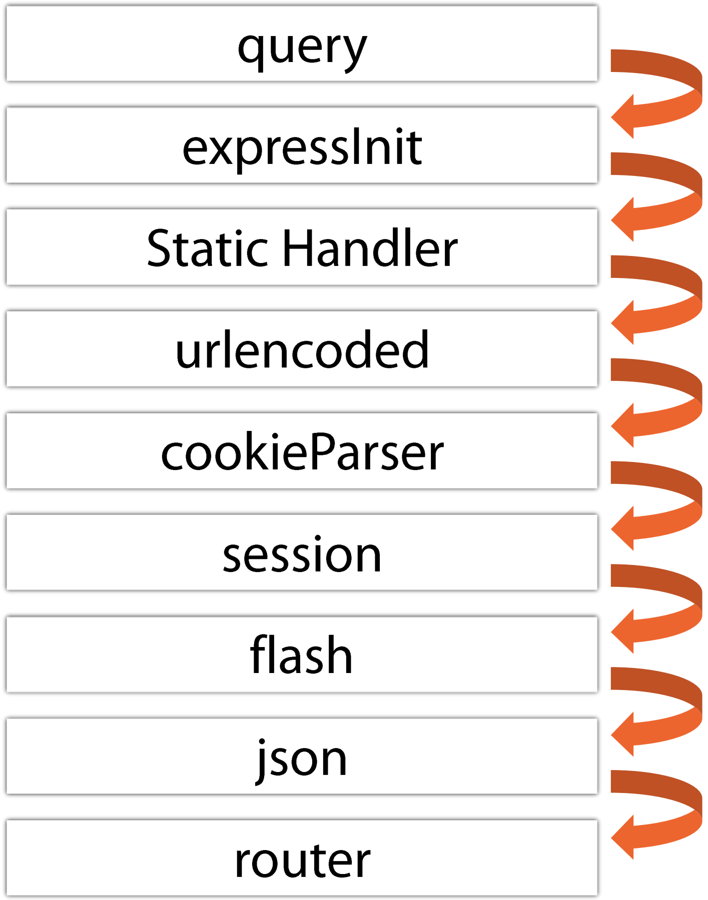
80
90
0

NODE.JS AND EXPRESS SUPPORTS MIDDLEWARE

SIMILAR TO ASP.NET PIPELINE OR ASP.NET MVC/WEB API FILTERS

Node.js and Express

```
// Implicit


app.use(express.static(__dirname + '/public'));

app.use(express.urlencoded());

app.use(express.cookieParser());

app.use(express.session({ secret: 'foo' }));

app.use(flash());

app.use(express.json());

// First causes ‘router’ to be added
app.get("/api/users", function (req, res) {");
  res.send({ firstName: "Hello" });
});
```



query

expressInit

Static Handler

urlencoded

cookieParser

session

flash

json

router

PASSPORT

Not that passport…Authentication Middleware

- Pluggable
  - Username/password
  - Facebook/Google/etc.
  - OAuth
- API Auth too

demonstration

CREATING THE REGISTER PAGE

demonstration

IMPLEMENTING THE REGISTER PAGE

demonstration

SECURING PASSWORDS

demonstration

INSTALLING PASSPORT

demonstration

VERIFYING CREDENTIALS

demonstration

INTEGRATING PASSPORT AND EXPRESS

demonstration

CREATE LOGIN PAGE

demonstration

AUTHORIZING PAGES AND APIS

Summary

- **Securing Node.js**
  - □ The pipeline in Express apps is simply a set of functions that are chained
  - □ Passport is security middleware that supports a variety of authentication schemes
  - □ For local security, you'll need to implement your own registration
  - □ Don't store passwords. Really. I am not kidding.
  - □ Passport is simple to plugin to Express but you need to implement verification and serialization
  - □ Once passport is plugged in, adding a login page is trivial
  - □ Testing for authentication can let you protect pages and APIs from anonymous users