

Universidad de La Laguna. Escuela Técnica Superior de Ingeniería Informática  
Tercero del Grado de Informática  
**SISTEMAS Y TECNOLOGÍAS WEB: 2ª PARTE**  
22/12/2016 2 páginas

Nombre: \_\_\_\_\_  
Fecha 27/10/2016\_\_\_\_\_

## HTTPS

1. ¿Cuales son las dos funcionalidades principales proveídas por la capa SSL?
2. ¿Es posible que alguien intercepte un mensaje utilizando HTTPS?
3. ¿Cuales son los tres objetivos de la fase de *handshake* entre un cliente y un servidor utilizando TLS?
4. ¿Como se llaman las tres fases en las que se puede descomponer la etapa de [handshake](#)?
5. Describa la primera fase del *handshake*
6. Describa la segunda fase del *handshake*
7. Describa la tercera fase del *handshake*
8. ¿Que tipo de cifrado se utiliza una vez que a finalizado con éxito la fase de handshake?
9. ¿Cuales son las dos razones por las que podríamos confiar en un certificado SSL?
10. Describa como funciona una firma digital
11. ¿Pueden en un coffee shop conocer los contenidos de mi tráfico HTTPS desde mi portátil sobre su red?
12. ¿Puede mi empresa conocer los contenidos de mi tráfico HTTPS sobre la red cuando uso la máquina que me proveen?

## Passport

1. ¿Que es OAuth?
2. ¿Quienes son los cuatro roles que aparecen en una autenticación con OAuth?
3. ¿Qué tres elementos de información suelen ser necesarios a la hora de registrar nuestra aplicación ante un proveedor de OAuth?
4. ¿Que se debe poner en *Redirect URI or Callback URL* cuando se está registrando nuestra aplicación?
5. Una vez que registramos la aplicación, el servicio provee las credenciales del cliente. ¿En que consisten esas credenciales?
6. ¿Que es *passport*, que funcionalidades provee y como funciona?
7. Rellene las partes que faltan:

```
var passport = require('passport');
var Strategy = require('_____').Strategy;
var github = require('octonode');
....
var datos_config = JSON.parse(JSON.stringify(config));

passport.use(new Strategy({
  clientID: datos_config.clientID,
  clientSecret: datos_config.clientSecret,
  callbackURL: callbackURL_
},
function(accessToken, refreshToken, profile, cb) {

  var token = datos_config.token;
  var client = github.client(_____);

  var ghorg = client.____('ULL-ESIT-SYTW-1617');

  ghorg._____(profile.username, (err,result) =>
  {
    if(err) console.log(err);
    if(result == true)
      return cb(null, profile);
  }
  )
});
```

```
        else {  
            return cb(null,null);  
        }  
    });  
}));
```