# AUTHENTICATON & AUTHORIZATION

*Kamel Rushaidat • Grand Circus Detroit*

# TOPICS

- What are Credentials?
- What is Authentication?
- What is Authorization?
- How to secure a Web application.
- The purpose of HTTPS
- How to configure Tomcat for Basic and Form authentication

# WHAT ARE CREDENTIALS

## Credentials

- Information used for identity verification[1]
- Usually consist of User name and Password

## Credential Categories/Factors[1]

- Knowledge factors (ex. Username, password or secret question)
- Possession factors (ex. Token, OpenID)
- Inherence factor (ex. Biometric data)

1. http://searchsecurity.techtarget.com/definition/authentication

# WHAT IS AUTHENTICATION?

## Authentication

- Authentication is the process of determining whether someone or something is, in fact, who or what it is declared to be.[1]

- Authentication is a process in which the credentials provided are compared to those on file in a database of authorized users' information on a local operating system or within an authentication server.[1]

- If the credentials match, the process is completed and the user is granted authorization for access.[1]

[1] http://searchsecurity.techtarget.com/definition/authentication

# WHAT IS AUTHORIZATION?

## Authorization

- Once a user is Authenticated they can be granted access to secure resources by Authorization

## Roles

- Roles define memberships with permission to access some secure resource (admin, user, guest or premium member)

## Users and Groups

- Users can have multiple roles and can be members of groups with specific roles

1. http://searchsecurity.techtarget.com/definition/authentication

# SECURING WEB APPLICATIONS

## Web Authentication Methods

- Basic
- Form
- Digest

2. https://docs.oracle.com/cd/E13222_01/wls/docs92/security/thin_client.html

# BASIC AUTHENTICATION

## Description

In Basic Authentication the Web server request that the client browser obtain the user credentials . The credentials are sent to the server for Authentication.[3]

## Precautions

Basic Authentication is not secure. User credentials are encoded but not encrypted when sent to the server.[3]

[3] http://java.boot.by/wcd-guide/ch05s03.html

# FORM AUTHENTICATION

## Description

- The server sends a custom login form to the client. The completed form is posted to the server by the client and authentication by the server. If authentication fails the client redirects to a custom error page.[3]

- Username and password form fields must be labeled j_username and j_password, respectively.

- Form action must be j_security_check (built into the servlet container).

## Precautions

Form Authentication is not secure. User credentials are submitted as plain text to the server.[3]

[3] http://java.boot.by/wcd-guide/ch05s03.html

# DIGEST AUTHENTICATION

## Description

Digest Authentication is similar to Basic Http authentication except that password is encrypted  before it is sent to the server[3]

[3] http://java.boot.by/wcd-guide/ch05s03.html

# HTTPS AND SSL

**SSL**

Secure Socket Layer – is an encrypted link between the client and server[4]

**HTTPS**

When the HTTP protocol is sent over SSL is is called HTTPS

**Applications**

Form and Basic authentication become more secure with the HTTP protocol

[4] https://www.digicert.com/ssl.htm

# OPEN ID

## Description

- OpenID is based on Oauth 2.0 which is a federated authorization mechanism[5]
- Third party providers authenticate users and provide access tokens for authorization to secured resources
- Third party providers such as Amazon, Google, Facebook, etc… provide OpenID APIs in various programming languages for use in securing Web applications

[5] http://openid.net/connect/faq/

# TOMCAT CONTAINER MANAGED SECURITY

## Authorization using Tomcat

Tomcat can be configured to support Basic and Form authentication and authorization

## Tomcat security configurations

- web.xml application deployment descriptor
- tomcat-users.xml
- Realms in the server.xml

# TOMCAT REALMS CONFIGUARATION

## Tomcat Realm[6]

- Specifies the database type used to store username and password credentials
- Realms also contain the list of roles used in authorization

## Tomcat Realms[6]

- JDBCRealm – Relational DB  via JDBC driver
- DataSourceRealm* - Relational DB  via JNDI JDBC DataSource
- JNDIRealm - LDAP based directory server, accessed via a JNDI provider
- UserDatabaseRealm - XML document (conf/tomcat-users.xml)
- MemoryRealm - stored in an in-memory object collection XML document (conf/tomcat-users.xml)
- JAASRealm - Java Authentication & Authorization Service (JAAS) framework

[6] https://tomcat.apache.org/tomcat-7.0-doc/realm-howto.html

# TOMCAT FORM AUTHENTICATION EXAMPLE

## Implementation[3]

1. Create login.jsp and error.jsp
2. Create Form element in login.jsp  with text and password input fields
3. Specify secure resource with <security-constraint> element in application web.xml
4. Specify authentication method with <login-config> element in application web.xml
5. Specify roles with <security-role> element in application web.xml

# RECAP

**What you should know at this point:**

- The difference between Authentication and Authorization.
- Credential categories.
- Web application Authentication methods.
- How to implement Basic and Form authentication