

AuthMatrix

Simplified Authorization Testing for
Web Applications



About Me – Mick Ayzenberg

- Security Engineer – 4 years
- Application Security – Web, Mobile, Desktop, SCADA, Embedded
- Specializations – Fuzzing, Crypto-currencies
- Lot of time spent pentesting web for large tech companies

Some Web PenTesting Insights

- Web Security Has Gotten Better
- Proper Design Decisions Prevent Most Issues
 - Patching Schedules
 - Framework/API Selection
- Testing Categories
 - Input Validation
 - Configuration
 - Authorization and Business Logic

Getting Easier to Do Right

- XSS
 - Testing: Fuzz Inputs, Evade Filters
 - Fixing: Encode all Outputs by Default
- SQLi
 - Testing: Fuzz Inputs, Evade Filters
 - Fixing: Parameterized Queries by Default
- CSRF, Authentication, Redirects
 - Testing: Identify them in the application
 - Fixing: Pick a good framework and let it do the work

The Problem

- Critical Bugs are Still Common
- Authorization in web applications and web services is still hard to get right!
 - Hard to develop
 - Hard to test
 - Unique per each application

Auth(n) vs. Auth(z)

- Authentication
 - Login
 - Session Management
- Authorization
 - Permissions
 - Access Controls

OWASP Top 10

OWASP Top 10 – 2013 (New)

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

OWASP Top 10 – Configuration

OWASP Top 10 – 2013 (New)

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

- Configuration
 - Patching
 - Easy for Scanners

OWASP Top 10 – Framework

OWASP Top 10 – 2013 (New)

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

- Configuration
 - Patching
 - Easy for Scanners
- Framework
 - Standard Testing Methodology
 - App-wide Fix

OWASP Top 10 – Specialized

OWASP Top 10 – 2013 (New)

A1 – Injection

A2 – Broken Authentication and Session Management

A3 – Cross-Site Scripting (XSS)

A4 – Insecure Direct Object References

A5 – Security Misconfiguration

A6 – Sensitive Data Exposure

A7 – Missing Function Level Access Control

A8 – Cross-Site Request Forgery (CSRF)

A9 – Using Known Vulnerable Components

A10 – Unvalidated Redirects and Forwards

- Configuration
 - Patching
 - Easy for Scanners
- Framework
 - App-wide Fix
 - Standard Testing Methodology
- Specialized
 - User/Permission Based
 - Unique to Each App

Authorization Examples

- <https://fake.com/GetUserInfo?id=53>
- <https://fake.com/Admin/ListAllUsers>
- <https://fake.com/Admin/DeleteUser?id=53>

Authorization Example

POST /Admin/DeleteUser HTTP/1.1

Host: fake.com

User-Agent: Mozilla/5.0

Content-Length: 8

id=53

Authorization Challenges

- Harder to fix within the framework
- Testing is time consuming
 - No generic testing conditions
 - Harder to automate
- Hard to repeat tests (regression)
 - Requires custom scripts

Authorization Testing

- Manual Process
 - Enumerate roles
 - Map entire application
 - Authenticate all necessary users
 - Test every combination
 - Run request
 - Observe results
 - Determine if behavior is correct
 - Record results
- Many Opportunities for Human Error

Threat Modeling

Action
Create
Read
Update
Delete

Frequency
Always
Sometimes
Never
Not Applicable

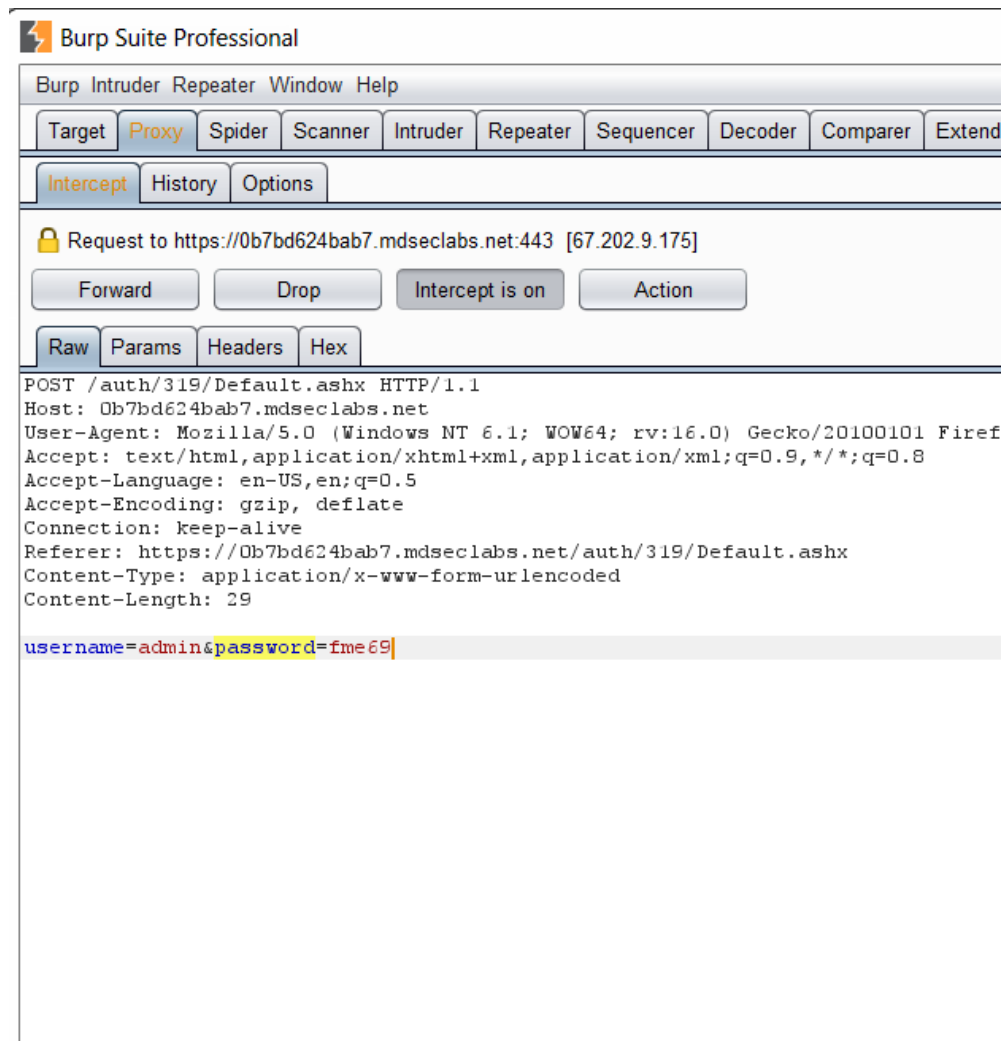
Asset	Action	Role			
		Anonymous	Employee	Manager	Admin
Employee Name	Create	Never	Never	Never	Always
	Read	Never	Sometimes	Always	Never
	Update	Never	Sometimes	Always	Never
	Delete	Never	Never	Never	Always
Employee SSN	Create	Never	Never	Never	Always
	Read	Never	Never	Never	Never
	Update	Never	Sometimes	Never	Never
	Delete	Never	Never	Never	Always
Employee Salary	Create	Never	Never	Never	Always
	Read	Never	Sometimes	Always	Never
	Update	Never	Never	Always	Never
	Delete	Never	Never	Never	Always

AuthMatrix – Design Goals

- Intuitive UI – similar to threat model table
- Manual condition mapping → Click and Run
- Reduce chances for human error
- Covers 90% of test scenarios
- Reproducible testing
- Make life easier (not harder)

Burp Suite

- Web App Testing Framework
- Commonly used
- Tamper/Request/Proxy individual HTTP messages



Burp Suite – Extender APIs

- Java, Python, or Ruby
- Features
 - Create new tabs within Burp Suite
 - Send new Requests
 - Modify Requests on the fly
 - Format data types
 - Scan results for specialized vulnerabilities

Burp Suite – Extensions

- API Signature headers
- HeartBleed scanning
- Integration with other tools
- Logging and note taking

Burp Suite - BApp Store

- Free extensions
 - Available from within Burp
- Several auth extensions
 - One request at a time
 - Manually review results
 - Confusing UI

AuthMatrix - DEMO

TargetProxySpiderScannerIntruderRepeaterSequencerDecoderComparerExtenderOptionsAlertsAuthMatrix

User	Session Token	(Optional) CSRF Token	Employee	HR	Manager	Anonymous
Anon	session_id=		<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
Employee 1	session_id=sOPklAZwzx9Zlx9zPmRw4zhZx7gjYivPh...		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
HR 1	session_id=/frWugmr1+Urqd1MlkrPsx+A4t9ibi+M...		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manager 1	session_id=1pQ1xirkf53f25/aMku/zkYyHrQhpg8vp...		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

ID	URL	Success Regex	Employee	HR	Manager	Anonymous
0	/HumanResources/admin/users/create	^HTTP/1\..1 200 OK	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
10	/HumanResources/timesheets	^HTTP/1\..1 200 OK	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
12	/HumanResources/manage/reviews	^HTTP/1\..1 200 OK	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
13	/HumanResources/manage/users	^HTTP/1\..1 200 OK	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

OriginalAnonEmployee 1HR 1Manager 1

RequestResponse

RawParamsHeadersHex

GET /HumanResources/manage/users HTTP/1.1
Host: ec2-52-34-220-246.us-west-2.compute.amazonaws.com
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10.10; rv:36.0) Gecko/20100101 Firefox/36.0
Accept: text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8
Accept-Language: en-US,en;q=0.5
Accept-Encoding: gzip, deflate
Referer: http://ec2-52-34-220-246.us-west-2.compute.amazonaws.com/HumanResources/manage
Connection: close
Cookie: session_id=sOPklAZwzx9Zlx9zPmRw4zhZx7gjYivPh7H3YW6njcg=

?<+>Type a search term0 matches

RunNew UserNew RoleSaveLoadClear

Revised Methodology

- Explore target application
- Add all functionality endpoints to AuthMatrix
- Define Roles and Users
- Fill out matrix tables
- Define success conditions (regex)
- Generate Session Tokens
- Run Save and Repeat

Usage Tips/Tricks

- Session token can be Header or Cookie
 - **Sessionid=abcd**
 - **Sessionid: abcd**
- Don't load untrusted configs
- Parameters matter in authorization as well

A4 - Insecure Direct Object Reference

The screenshot displays a web application security tool interface. At the top, there is a table listing users and their session tokens, with checkboxes for 'Employee', 'Manager', and 'Only Employee 10' roles.

User	Session Token	(Optional) CSRF Token	Employee	Manager	Only Employee 10
Employee 5	session_id=m4qXkWv5QzstNdISBVfzgq+JjXPGriEW/...		<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Manager 1	session_id=J50lhiv+rIZnVoF5XIEPJYj9QJG5u086l/7J...		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Employee 10	session_id=/ClubUgm/QZMWK1/J6TzqPTs41+Fc/x...		<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

Below this table is another table showing request logs with columns for ID, Request Name, Success Regex, and role checkboxes.

ID	Request Name	Success Regex	Employee	Manager	Only Employee 10
55	/HumanResources/users/10/preferences	^HTTP/1\1 200 OK	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>
58	/HumanResources/users/10/profile	^HTTP/1\1 200 OK	<input type="checkbox"/>	<input type="checkbox"/>	<input checked="" type="checkbox"/>

The interface includes tabs for 'Original', 'Employee 5', 'Manager 1', and 'Employee 10'. Below these are tabs for 'Request' and 'Response'. The 'Response' tab is active, showing a raw HTTP response.

Raw Headers Hex HTML Render

```
HTTP/1.1 200 OK
Server: Apache-Coyote/1.1
Content-Type: text/html; charset=utf-8
Date: Fri, 22 Jan 2016 01:34:29 GMT
Connection: close
Content-Length: 6670

<!DOCTYPE html>
<html class="no-js" lang="en">
<head>
<title>HR User Preferences / 1111</title>
```

At the bottom, there is a search bar with the text 'Type a search term' and a '0 matches' indicator. Below the search bar are buttons for 'Run', 'New User', 'New Role', 'Save', 'Load', and 'Clear'.

Current Limitations

- Only static CSRF tokens supported currently
- No support for multiple session tokens
- Success detection requires regex
- Web service signature support
 - Though can often be coupled with other extensions
- Permission Exceptions
 - “Admins can, Managers can, but Admin-Managers can not.”

Whats Next?

- Open sourced soon...
 - <https://github.com/SecurityInnovation/AuthMatrix>
- Explore Burp session management integration
 - Macros to renew session tokens automatically
 - Macros to update CSRF tokens
- Submission to BApp Store for free download



QUESTIONS?