Security Testing Course Handout

Gauntlt

- Home page: http://gauntlt.org
- Source: http://github.com/gauntlt/gauntlt
- XML parsing example: https://github.com/gauntlt/gauntlt/blob/master/examples/nmap/xml_output.attack
- Using environment variables in your attacks: https://github.com/gauntlt/gauntlt/wiki/Using-Cucumber-Profiles-and-Environment-Variables-with-Gauntlt

Gauntlt Attack Examples

- Gauntlt main repo: https://github.com/gauntlt/gauntlt/tree/master/examples
- Gauntlt-demo repo: https://github.com/gauntlt/gauntlt-demo/tree/master/examples
-

Course Labs

- Security testing class repo: https://github.com/wickett/security-testing-class
- Gauntlt-Docker: http://github.com/gauntlt/gauntlt

Arachni

- Home page: http://www.arachni-scanner.com/

Fuzzing with DIRB

- Source: https://sourceforge.net/projects/dirb/
- Tutorial: http://www.slyth-sec.com/dirb-tutorial/

Nmap

- Home page: http://nmap.org
- Nmap scripting engine: https://nmap.org/book/nse-scripts-list.html
- Nmap testing WordPress: https://blog.wpscans.com/wordpress-vulnerability-testing-wordpress-with-nmap/
- Nmap book: https://nmap.org/book/

DevOps Classes in Lynda and LinkedIn Learning Library

- *DevOps Foundations*
- *DevOps Foundations: Continuous Delivery/Continuous Integration*
- *DevOps Foundations: Infrastructure Automation*

Book Recommendations

- *Agile Application Security* by Jim Bird, Laura Bell, Michael Burton-Spall, and Rich Smith (There is a chapter focused on Gauntlt, but the book is really good all around.)
- *Continuous Delivery* by David Farley and Jez Humble (Not a security-specific book, but lots of good practices)

Articles on Gauntlt

https://www.kainos.com/closer-look-gauntlt/

Articles on Security Testing

https://www.kainos.com/bringing-security-pipeline/

Gauntlt Cheat Sheet

Gherkin Keywords

- Feature
- Scenario
- Given
- When
- Then
- And
- But
- Background

Other Gherkin Operators

- """ - Triple Quote used for document strings
- | - Pipe used for creating tables
- @ - Used for making tags
- # - Adding unexecuted comments

Gauntlt Tags

- No tag - The default time allowed per scenario is 3s
- @slow - Allows 30s per scenario
- @reallyslow - Allows 10 minute execution per scenario

Gauntlt Options

-t, --tags=<s> Only execute specified tags

-l, --list List defined attacks

-s, --steps List the gauntlt step definitions that can be used inside of attack files

-a, --allsteps List all available step definitions including aruba step definitions which help with file and parsing operations

-f, --format=<s> Available formats: html, json, junit, progress

-v, --version Print version and exit

-h, --help Show this message

Common Gauntlt Output Parsing Steps

/^the output from "([^"]*)" should contain "([^"]*)"$/

/^the output from "([^"]*)" should not contain "([^"]*)"$/

/^the output should contain "([^"]*)"$/

/^the output should contain:$/

/^the output should match \/([^\/]*)\/$/

/^the output should match:$/

/^the output should not contain "([^"]*)"$/

/^the output should not contain:$/

/^the output should not match \/([^\/]*)\/$/

/^the output should not match:$/

/^the output(?: from "(.*?)")? should contain exactly "(.*?)"$/

/^the output(?: from "(.*?)")? should contain exactly:$/

/^the stderr from "([^"]*)" should not contain "([^"]*)"$/

/^the stderr should not contain "([^"]*)"$/

/^the stderr should not contain:$/

/^the stderr(?: from "(.*?)")? should contain(exactly)? "(.*?)"$/

/^the stderr(?: from "(.*?)")? should contain(exactly)?:$/

/^the stdout from "([^"]*)" should not contain "([^"]*)"$/

/^the stdout should not contain "([^"]*)"$/

/^the stdout should not contain:$/

/^the stdout(?: from "(.*?)")? should contain(exactly)? "(.*?)"$/

/^the stdout(?: from "(.*?)")? should contain(exactly)?:$/

Gauntlt Attack Steps

/^"(\w+)" is installed in my path$/

/^"Heartbleed" is installed$/

/^"arachni" is installed$/

/^"curl" is installed$/

/^"dirb" is installed$/

/^"garmr" is installed$/

/^"nmap" is installed$/

/^"sqlmap" is installed$/

/^"sslyze" is installed$/

/^I launch (?:a|an) "Heartbleed" attack with:$/

/^I launch (?:a|an) "arachni" attack with:$/

/^I launch (?:a|an) "arachni-(.*?)" attack$/

/^I launch (?:a|an) "curl" attack with:$/

/^I launch (?:a|an) "dirb" attack with:$/

/^I launch (?:a|an) "garmr" attack with:$/

/^I launch (?:a|an) "generic" attack with:$/

/^I launch (?:a|an) "nmap" attack with:$/

/^I launch (?:a|an) "nmap-(.*?)" attack$/

/^I launch (?:a|an) "sqlmap" attack with:$/

/^I launch (?:a|an) "sslyze" attack with:$/

/^the "(.*?)" command line binary is installed$/

/^the DIRB_WORDLISTS environment variable is set$/

/^the file "(.*?)" should contain XML:$/

/^the file "(.*?)" should not contain XML:$/

/^the following cookies should be received:$/

/^the following environment variables:$/

/^the following profile:$/