Server Side Security Tools

Preetika Soni (41751203116)
Bhavya Takkar (35251203116)
Vishal Dubey (03451203116)
Harsh Jain (01151203116)
Udit Dahima (03351203116)

Server-Side refers to operations that are performed by the server in client-server relationship in a Computer network .

When protecting your website against hacks or accidental file corruption, there are two main areas to consider.
First, you'll need to secure your actual website. However, your site's server is another potential access point, which hackers can use to locate personal information and cause major issues (such as file corruption).
Therefore, by securing your server in addition to your website, you can vastly improve your website's functionality and your visitors' viewing experience.

Server side security tools

- WEB APPLICATION FIREWALLS (WAFs)
- FUZZERS

WEB APPLICATION FIREWALLS (WAFs)

Web Application firewalls (WAFs)

- A Web Application Firewall (WAF) detects web traffic looking for suspicious activity; it can then automatically filter out illegitimate traffic based on rule sets that you ask it to apply.
- By controlling its input and output and the access to and from the application.
- Running as and server plug in or cloud -based service , a WAF inspects every html, https etc and data packets.
- A WAF is also able to detect and prevent new unknown attacks by watching for unfamiliar patterns in the traffic data.

Different types of WAFs

- Network-based WAFs
- Host- based WAFs
- Cloud- based WAFs

Web Application Firewall (WAFs) protects against

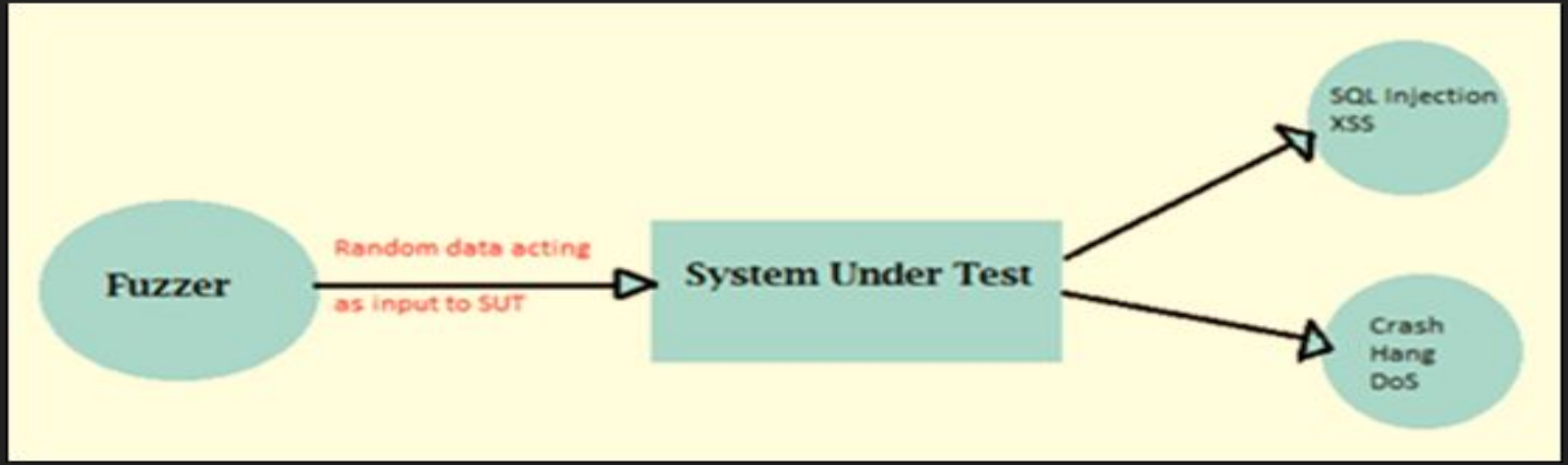
- SQL injection , comment spam
- Cross- site scripting (XSS)
- Distributed denial of service (DDoS) attacks.
- Application- specific attacks (WordPress , Core Commerce) and many more.

FUZZERS

What is fuzzing?

A kind of random testing where inputs to a case are generated randomly.

After which the system is monitored for various exceptions.



GOAL

Prevention of these security vulnerabilities

- **Crashes** due to memory error or uncaught exception
- Hangs due to Non termination
- Failing built-in code
- Incorrect output

Used to discover

- Coding errors
- Security loopholes in software, operating systems, or networks

Kinds of Fuzzing

Black box

The tool knows nothing about the program or its input .

Easy to use and get started, but will explore only **shallow states** unless it gets lucky

Grammar based

The tool generates input informed by a grammar.

More work to use, to produce the grammar. but can go deeper in the state space.

White box

The tool generates new inputs at least partially informed by the code of the program being fuzzed.

How to do Fuzz Testing

The steps for fuzzy testing include the basic testing steps :

Step 1) Identify the target system.

Step 2) Identify inputs.

Step 3) Generate Fuzzed data.

Step 4) Execute the test using fuzzy data.

Step 5) Monitor system behavior.

Step 6) Log defects.

Types of bugs detected by Fuzz Testing

- Assertion failures and memory leaks.
- Invalid input.
- Correctness bugs.

Examples of Fuzzers

- **Mutation-Based Fuzzers**

  It alters existing data samples to create new test data.

- **Generation-Based Fuzzers**

  It defines new data based on the input of the model. It starts generating input from the scratch based on the specification.

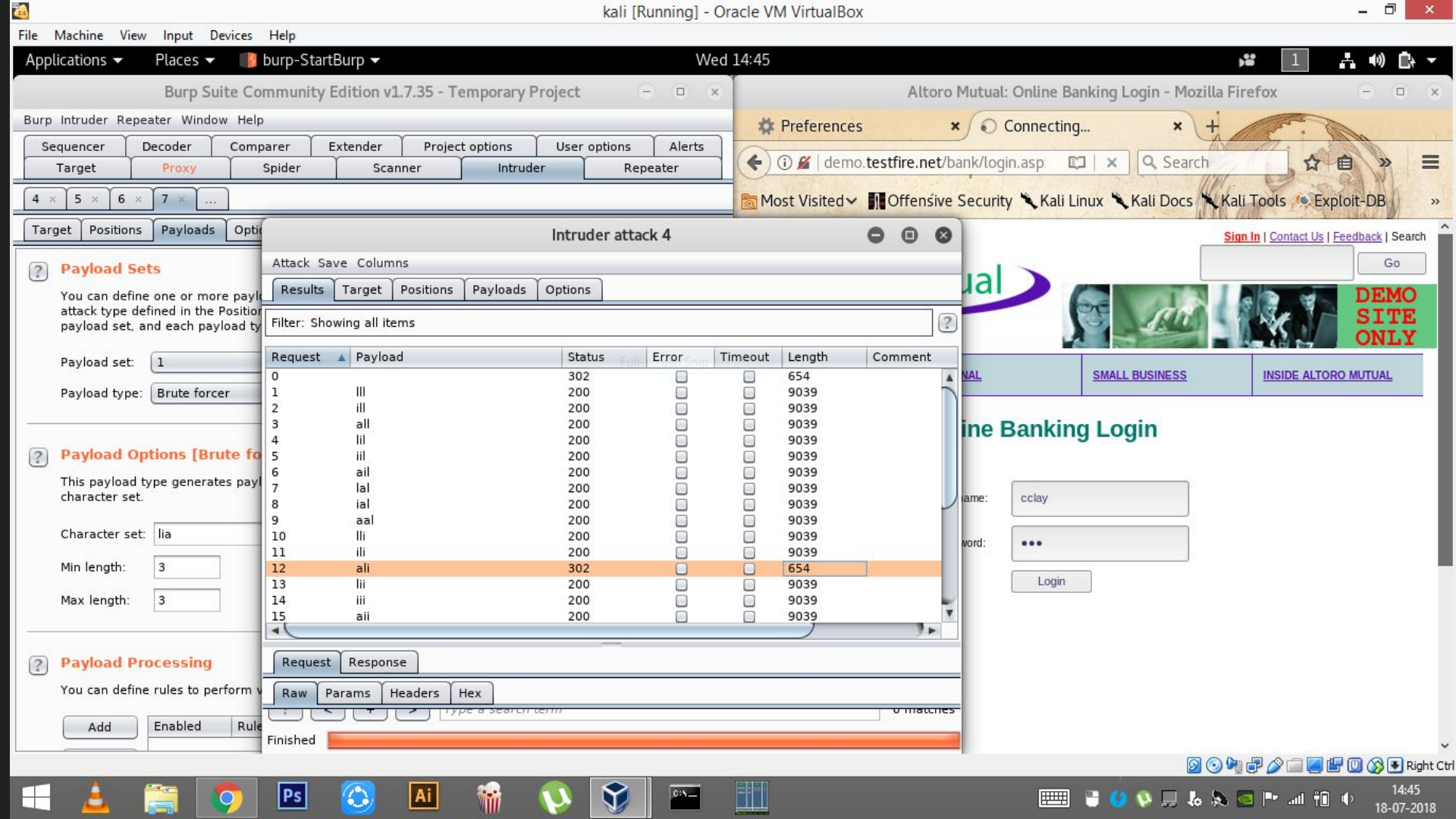
- **Protocol-Based Fuzzers**

  Detailed knowledge of protocol format is being tested .This is also known as syntax testing, grammar testing & robustness testing.

Why to do Fuzz Testing?

- Usually, Fuzzy testing finds the most serious security fault or defect.
- Fuzz testing gives more effective result when used with Black Box Testing, Beta Testing, and other debugging methods.
- Fuzz testing is used to check the Vulnerability of software. It is very cost effective testing techniques.
- Fuzz testing is one of the black box testing technique. Fuzzing is one of the most common method hackers used to find vulnerability of the system.

Fuzzing Web Applications with Burp-Suite Intruder Tool


kali [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Applications Places burp-StartBurp Wed 14:45
Burp Suite Community Edition v1.7.35 - Temporary Project
Burp Intruder Repeater Window Help
Sequencer Decoder Comparer Extender Project options User options Alerts
Target Proxy Spider Scanner Intruder Repeater
4 5 6 7 ...
Target Positions Payloads Optic
Payload Sets
You can define one or more payl
attack type defined in the Positior
payload set, and each payload ty
Payload set: 1
Payload type: Brute forcer
Payload Options [Brute fo
This payload type generates payl
character set.
Character set: lia
Min length: 3
Max length: 3
Payload Processing
You can define rules to perform v
Add
Enabled
Rule
Intruder attack 4
Attack Save Columns
Results Target Positions Payloads Options
Filter: Showing all items
Request Payload Status Error Timeout Length Comment
0 302 654
1 lll 200 9039
2 ill 200 9039
3 all 200 9039
4 lil 200 9039
5 iil 200 9039
6 ail 200 9039
7 lal 200 9039
8 ial 200 9039
9 aal 200 9039
10 lli 200 9039
11 ili 200 9039
12 ali 302 654
13 lii 200 9039
14 iii 200 9039
15 aii 200 9039
Request Response
Raw Params Headers Hex
Finished
Altoro Mutual: Online Banking Login - Mozilla Firefox
Preferences
Connecting...
demo.testfire.net/bank/login.asp
Search
Most Visited Offensive Security Kali Linux Kali Docs Kali Tools Exploit-DB
Sign In | Contact Us | Feedback | Search
Go
DEMO SITE ONLY
SMALL BUSINESS
INSIDE ALTORO MUTUAL
ine Banking Login
ame: cclay
word: •••
Login
Right Ctrl
14:45
18-07-2018

Advantages of Fuzz Testing

- Fuzz testing improves software Security Testing.
- Bugs found in fuzzing are sometimes severe and most of the time used by hackers including crashes, memory leak, unhandled exception, etc.
- If any of the bugs fail to get noticed by the testers due to the limitation of time and resources those bugs are also found in Fuzz testing.

Disadvantages of Fuzz Testing

- Fuzz testing alone cannot provide a complete picture of an overall security threat or bugs.
- Fuzz testing is less effective for dealing with security threats that do not cause program crashes, such as some viruses, worms, Trojan, etc.
- Fuzz testing can detect only simple faults or threats.
- To perform effectively, it will require significant time.
- Setting a boundary value condition with random inputs is very problematic but now using deterministic algorithms based on users inputs most of the testers solve this problem.

Summary

In Software Engineering, Fuzz testing shows the presence of bugs in an application.

Fuzzing cannot guarantee detection of bugs completely in an application.

But by using Fuzz technique, it ensures that the application is robust and secure, as this technique helps to expose most of the common vulnerabilities.

Thank you!