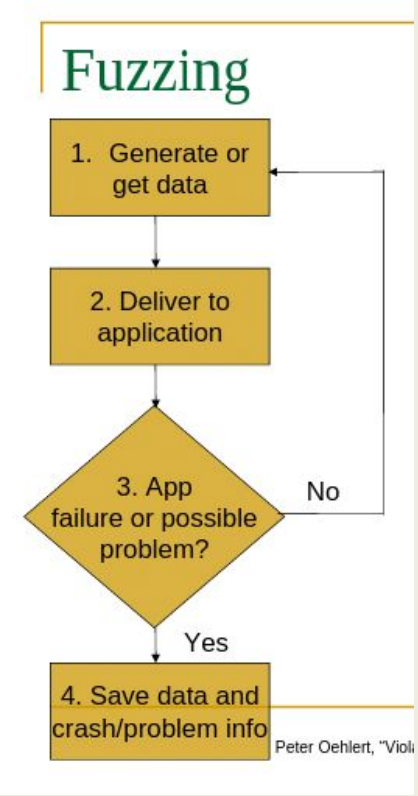
Evolutionary Fuzzing

Presentation 1
Introduction and structure of a simple fuzzer.

Bofin Babu
2013H313085

Fuzzing


Fuzzing
1. Generate or get data
2. Deliver to application
3. App failure or possible problem?
No
Yes
4. Save data and crash/problem info
Peter Oehlert, "Viol

A simple fuzzer

```
import socket

buffer = ["A"]
counter = 2

while len(buffer) <= 30:
        buffer.append("A"*counter)
        counter = counter+100

commands = ["MKD","GET","STOR","SYST", "XYZS"]

for command in commands:
        for string in buffer:
                print "Sending the "+command+" command with "+ str(len(string))+" bytes."
                s = socket.socket(socket.AF_INET, socket.SOCK_STREAM)
                connect=s.connect(('172.16.6.133',21))
                s.recv(1024)
                s.send('USER ftp\r\n')
                s.recv(1024)
                s.send(command+' '+string+'\r\n')
                s.recv(1024)
                s.send('QUIT ftp \r\n')
                s.close()
```

Genetic Algorithm (GA)

Uses techniques inspired by natural evolution such as,

- **Inheritance**
  - The ability of modeled objects to mate, mutate and propagate their problem solving genes to the next generation, in order to produce an evolved solution to a particular problem.
  - The selection of objects that will be inherited from in each successive generation is determined by a fitness function.
- **Mutation**
  - Mutation alters one or more gene values in a chromosome from its initial state
  - Used to maintain genetic diversity from one generation of a population of genetic algorithm chromosomes to the next.
- **Selection**
  - The stage of a genetic algorithm in which individual genomes are chosen from a population for later breeding (using crossover operator).
- **Crossover**
  - A genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next.
  - Analogous to reproduction - taking more than one parent solutions and producing a child solution from them.

GA & fuzzing

- Evolutionary Testing uses evolutionary algorithms to search for software test data
- 1 individual = 1 application input. A population of individuals are evolved according to their fitness score.