Loadtest und Stresstest

Thomas Antensteiner, Philipp Auinger

Was ist ein Loadtest?

- Ein Test in dem eine gewisse Anzahl von Abfragen gleichzeitig getätigt werden um zu sehen ob das System diese standhält.

Quelle: https://www.loadview-testing.com/load-testing-vs-stress-testing

Was ist ein Stresstest?

- Erhöht die Anzahl der Anfragen bis das System langsamer wird
  - Oft wird auch bis zum Zusammenbruch des Systems getestet

Quelle: https://www.loadview-testing.com/load-testing-vs-stress-testing

Gatling

Gatling

- Scala
- Gatling Corp
- Loadtest REST-Endpoints/Websites

- HTML reports
- viele User gleichzeitig simulieren
- verschiedene User, verschiedene Dinge machen lassen
- User mit Daten aus File versorgen

Gatling Recorder - Configuration

Gatling

Recorder mode

HTTP Proxy

Network

Listening port*: localhost HTTP/HTTPS 8000 HTTPS mode: Self-signed Certificate

Outgoing proxy: host: HTTP HTTPS Username Password

Simulation Information

Package: Class Name*: RecordedSimulation

- [ ] Follow Redirects?
- [x] Infer HTML resources?
- [x] Automatic Referers?
- [x] Remove cache headers?
- [ ] Use Class Name as request prefix?
- [ ] Save & check response bodies?

Output

Simulations folder*: C:\opt\gatling-charts-highcharts-bundle-3.3.1\user-files\simulations Browse

Encoding: Unicode (UTF-8)

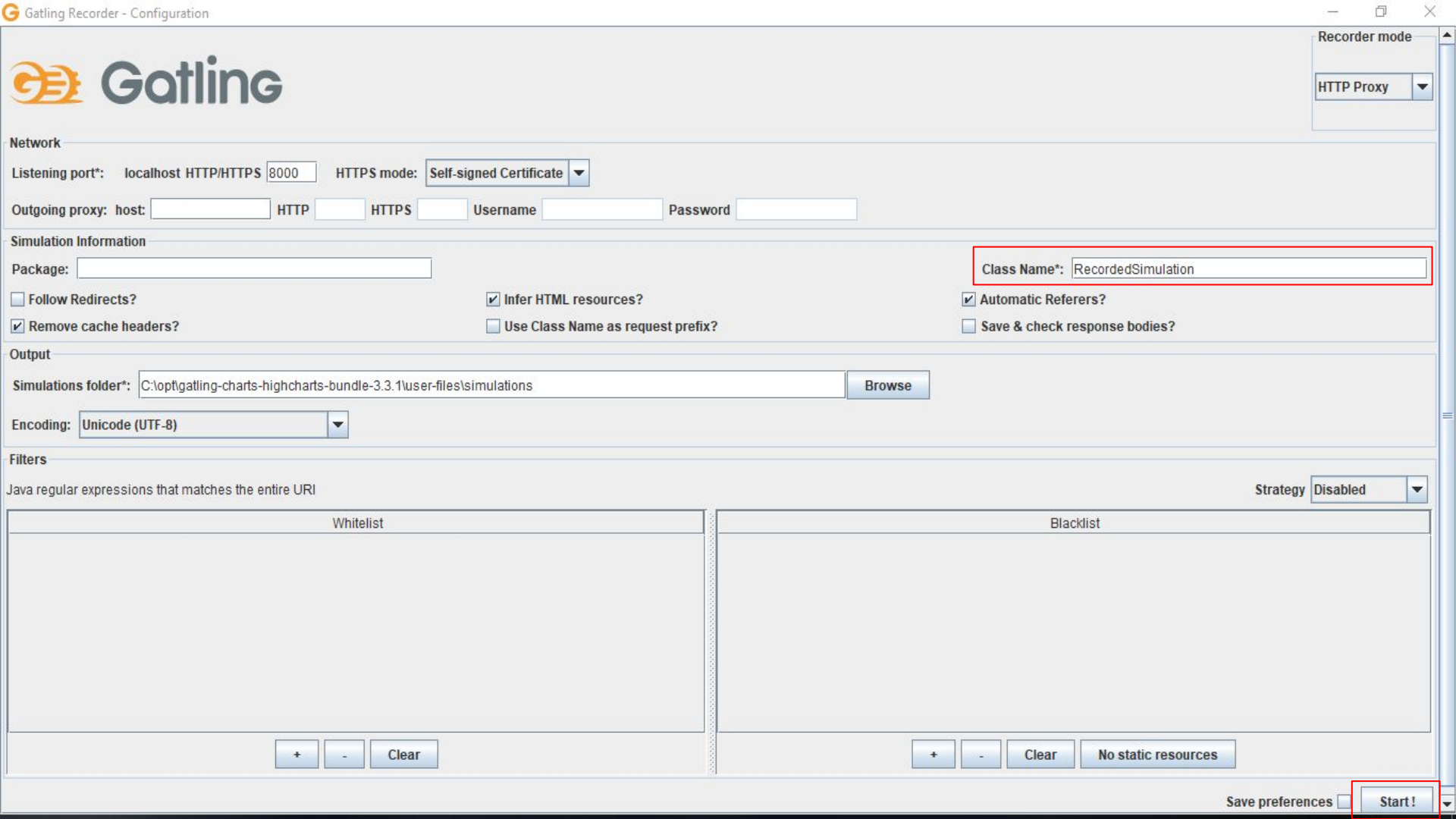
Filters

Java regular expressions that matches the entire URI

Strategy Disabled

Whitelist	Blacklist
+ - Clear	+ - Clear No static resources

Save preferences Start!


Gatling Recorder - Configuration
Gatling
Recorder mode
HTTP Proxy
Network
Listening port*: localhost HTTP/HTTPS 8000 HTTPS mode: Self-signed Certificate
Outgoing proxy: host: HTTP HTTPS Username Password
Simulation Information
Package:
Class Name*: RecordedSimulation
Follow Redirects?
Infer HTML resources?
Automatic Referers?
Remove cache headers?
Use Class Name as request prefix?
Save & check response bodies?
Output
Simulations folder*: C:\opt\gatling-charts-highcharts-bundle-3.3.1\user-files\simulations
Browse
Encoding: Unicode (UTF-8)
Filters
Java regular expressions that matches the entire URI
Strategy Disabled
Whitelist
Blacklist
+
-
Clear
+
-
Clear
No static resources
Save preferences
Start !

Gatling

```
class Example extends Simulation{
  val httpProtocol = http
    .baseURL("http://localhost:8080")

  val mySzenario = Szenario("my Szenario").exec(...)

  setUp(
    mySzenario.inject(
      constantUsersPerSec(40) during (20 minutes)
    )
  ).protocols(httpProtocol)
}
```

Gatling

```
class Example extends Simulation{
  val httpProtocol = http
    .baseURL("http://localhost:8080")

  val mySzenario = Szenario("my Szenario").exec(...)

  setUp(
    mySzenario.inject(
      constantUsersPerSec(40) during (20 minutes)
    )
  ).protocols(httpProtocol)
}
```

```
val httpProtocol = http
    .baseUrl("http://computer-database.gatling.io")
    .acceptHeader("text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8")
    .doNotTrackHeader("1")
    .acceptLanguageHeader("en-US,en;q=0.5")
    .acceptEncodingHeader("gzip, deflate")
    .userAgentHeader("Mozilla/5.0 (Windows NT 5.1; rv:31.0) Gecko/20100101 Firefox/31.0")
```

Gatling

```
class Example extends Simulation{
  val httpProtocol = http
    .baseURL("http://localhost:8080")

  val mySzenario = Szenario("my Szenario").exec(...)

  setUp(
    mySzenario.inject(
      constantUsersPerSec(40) during (20 minutes)
    )
  ).protocols(httpProtocol)
}
```

```
val search = exec(http("Home")
    .get("/"))
    .pause(7)
    .exec(http("Search")
      .get("/computers?f=macbook"))
    .pause(2)
    .exec(http("Select")
      .get("/computers/6"))
    .pause(3)
```

Gatling

```
class Example extends Simulation{
  val httpProtocol = http
    .baseURL("http://localhost:8080")

  val myScenario = Scenario("my Scenario").exec(...)

  setUp(
    myScenario.inject(
      constantUsersPerSec(40) during (20 minutes)
    )
  ).protocols(httpProtocol)
}
```

Kontrolle der Anzahl der User

```
atOnceUsers
rampUsers
constantUsersPerSec
rampUsersPerSec
heavisideUsers
nothingFor
incrementUsersPerSec          New!
```

JMeter

- Fähigkeit verschiedenste applications/server/protocol Typen zu testen:
  - Web - HTTP, HTTPS (Java, NodeJS, PHP, ASP.NET, …)
  - SOAP / REST Webservices
  - FTP
  - Database via JDBC
  - LDAP
  - Message-oriented middleware (MOM) via JMS
  - Mail - SMTP(S), POP3(S) and IMAP(S)
  - Native commands or shell scripts
  - TCP
  - Java Objects
- JMeter eignet sich besonders beim Testen von REST Services.

Quelle: http://jmeter.apache.org, https://de.wikipedia.org/wiki/Apache_JMeter

JMeter

- GUI - Tool
- JMeter Intellij Plugin

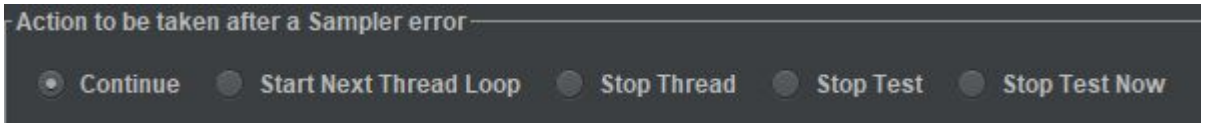
JMeter MQTT Plugin

- https://github.com/emqx/mqtt-jmeter
- “MQTT - Samplers”

Quelle: https://www.youtube.com/watch?v=M-iAXz8vs48, https://www.youtube.com/watch?v=8loLHbhfyh0

JMeter - Begriffe

- Test-Plan: Ein Container wo mehrere Test-Elemente / Schritte zusammengefasst werden
- Thread-Group: “Users” die diesen Test durchführen werden
  - Was soll bei einem Fehler gemacht werden:



- Zur Ausgabe der Ergebnisse “Listener hinzufügen”
- Ramp-Up Period: Zeit um auf die Nutzerzahl zu kommen

Quelle: https://www.youtube.com/watch?v=M-iAXz8vs48, https://www.youtube.com/watch?v=8loLHbhfyh0

JMeter - Command Line

- Non GUI - Mode
- jmeter -n -t HTL-Leonding-Website.jmx -l result.csv

- Ergebnisse werden in .CSV File geschrieben

```
C:\opt\JMeter\apache-jmeter-5.2.1\bin>jmeter -n -t HTL-Leonding-Website.jmx -l result.csv
Creating summariser <summary>
Created the tree successfully using HTL-Leonding-Website.jmx
Starting standalone test @ Wed Feb 05 17:16:08 CET 2020 (1580919368905)
Waiting for possible Shutdown/StopTestNow/HeapDump/ThreadDump message on port 4445
summary =      1 in 00:00:02 =    0.7/s Avg:  1318 Min:  1318 Max:  1318 Err:     0 (0.00%)
Tidying up ...    @ Wed Feb 05 17:16:10 CET 2020 (1580919370970)
... end of run
```

Quelle: https://www.youtube.com/watch?v=K26q5VgwLKk&list=PLhW3qG5bs-L-zox1h3eIL7CZh5zJmci4c&index=9

DANKE FÜR IHRE AUFMERKSAMKEIT