Bot or Not?

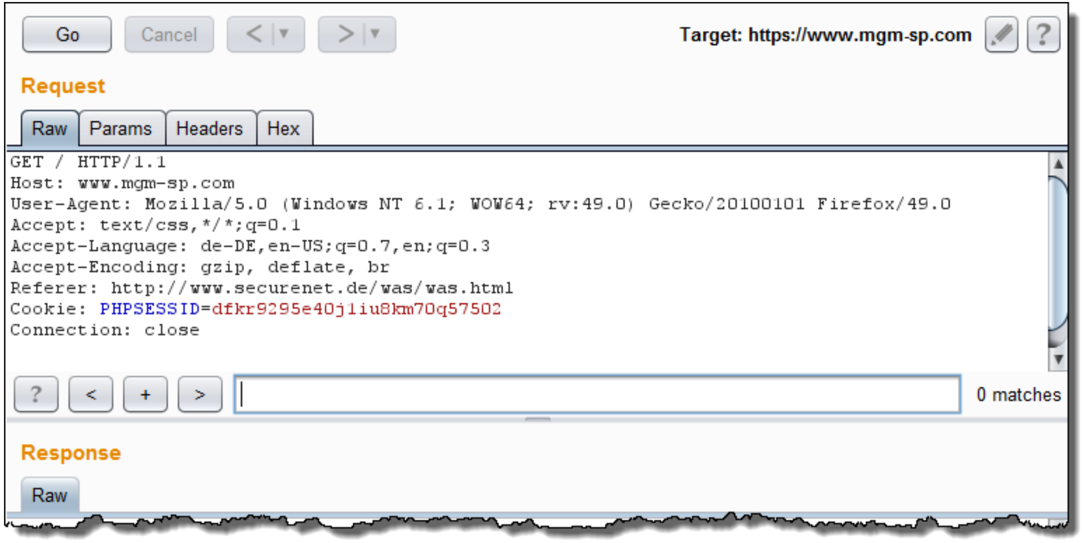
Mitigating Automated Threats to Web Applications

Bastian Braun
mgm security partners

21 November 2017
OWASP Stammtisch München

Background: Automation in the Web

- web communication = requests + responses
- stateless HTTP allows uncontrolled repetitions of previous requests



```
GET / HTTP/1.1
Host: www.mgm-sp.com
User-Agent: Mozilla/5.0 (Windows NT 6.1; WOW64; rv:49.0) Gecko/20100101 Firefox/49.0
Accept: text/css,*/*;q=0.1
Accept-Language: de-DE,en-US;q=0.7,en;q=0.3
Accept-Encoding: gzip, deflate, br
Referer: http://www.securenet.de/was/was.html
Cookie: PHPSESSID=dfkr9295e40j1iu8km70q57502
Connection: close
```

Background: Automation in the Web

- practical
  - easily expandable
  - more robust / fail safe than stateful communication
  - business logic scalable & movable (see Angular, React, …)
- problematic
  - (in-)secure workflows
    - control-flow integrity
  - automated actions

Threats by Automation

- registration
  - e.g. email accounts for spammers, newsletters, username enumeration
- login
  - e.g. password brute-forcing, user lock-out
- password reset
  - e.g. email flooding, username enumeration
- parameterized search queries
  - data harvesting

Detection

- depends on feature logic
- approaches
    - detect massive requests from same IP
        - requires threshold → evade by spreading
    - generate client fingerprint to identify source
        - no fingerprint → suspicious
        - spoofed fingerprints → sanity check
    - device cookies
    - require authentication (login) before granting access
        - protect registration & login

Countermeasures: Theory

- CAPTCHAs
- additional knowledge
- tarpit
- SMS TANs
- proof-of-work systems
- IP locks
- user locks

Countermeasures: Practice

Countermeasure	Practical Issues
CAPTCHAs	annoying, bad usability, breakable
additional knowledge	annoying
tarpit	susceptible to DoS attacks, temporary user lockout
SMS TANs	automated triggers
proof-of-work systems	hard to scale
IP locks	false positives / collateral damage if NAT
user locks	massive user-lock out

Countermeasures: Applicability

Functionality	Appropriate Detection	Applicable Anti-Automation	Unsuitable Approaches
Registration	Client IP, Client Fingerprint	CAPTCHA, Proof-of-Work, IP Locks	Additional Knowledge, Tarpit, SMS TAN, User Locks
Password Reset	Client IP, Client Fingerprint, Device Cookie	CAPTCHA, Additional Knowledge, SMS TAN, Proof-of-Work, IP Locks	Tarpit, User Locks
Login	Client IP, Client Fingerprint, Device Cookie	Additional Knowledge, Tarpit, SMS TAN, Proof-of-Work, IP Locks, User Locks	CAPTCHA
Contact Form	Client IP, Client Fingerprint (Device Cookie, Authentication)	CAPTCHA, Proof-of-Work, IP Locks	Additional Knowledge, Tarpit, SMS TAN, User Locks
Newsletter Registration	Client IP, Client Fingerprint, Device Cookie (Authentication)	CAPTCHA, Proof-of-Work, IP Locks	Additional Knowledge, Tarpit, SMS TAN, User Locks
Parameterized Search Queries	Client IP, Client Fingerprint, Device Cookie, Authentication	Proof-of-Work, IP Locks	CAPTCHA, Additional Knowledge, Tarpit, SMS TAN, User Locks

Conclusion

- open issues
    - how to protect machine-2-machine APIs?
    - how to distinguish competitors from Google?
        - e.g. prevent automatic price analysis by competitors vs give Google crawler access to portfolio
- user acceptance still the biggest problem
- awareness during development processes often low