



Remo Presentation

(Positive ModSecurity Rulesets / Input Validation)

Dr. Christian Folini
netnea.com / Swiss Post
christian.folini@netnea.com

OWASP

Copyright © The OWASP Foundation
Permission is granted to copy, distribute and/or modify this document
under the terms of the OWASP License.

Who am I and how did I get here?

- I started out in IT as an innocent medievalist
- Then I got interested in Apache
- Then I wanted to secure my servers with ModSecurity
- Then things got a bit out of hand



Defense Preparations

- Trying to build a strong defense position with ModSecurity
- Import cool rulesets (gotroot?)
- It felt like making a basket waterproof
- Marcus Ranum:
“Default Permit is the number one stupid thing in Computer Security”



Negative Rulesets

You patch against all known attacks.
That's like basic spam filtering and it boils down to *Default Permit*.

More from Marcus: “Systems based on *Default Permit* are the computer security equivalent of empty calories: tasty, yet fattening.”

And given the gotroot ruleset has 10000+ rules, you'll agree on the fattening effect.

Generic Negative Rulesets

- Not patching against individual attacks but against generic patterns
- Official Core-Rules: That's about as good as you can get in my humble opinion

```
SecRule REQUEST_FILENAME|ARGS "\b(?:user_(?:(?object|table|user)s|password|group)|a(?:tt(?:rel|typ)id|ll_objects)|object_(?:(?nam|typ)e|id)|pg_(?:attribute|class)|column_(?:name|id)|substr(?:ing)?|table_name|mb_users|rownum)\b" \
"capture,t:htmlEntityDecode,t:lowercase,t:replaceComments,ctl:auditLogParts=
+E,log,auditlog,msg:'SQL Injection Attack. Matched signature <{%TX.0}>',id:'950906',severity:'2'"
```

Positive Rulesets

- **POST /action/submit.php**
 - Headers
 - Host: `mail.companyx.com`
 - Referer: `.{0,256}`
 - User-Agent: `.{0,256}`
 - Postparameters
 - username: `[0-9a-zA-Z]{4,16}`
 - password: `.{0,16}`
 - submit: `login`

The positive rule from Remo is translated into a whitelist ModSecurity rule. This means, that you have to define the good arguments in Remo. Requests with arguments, that do not match this positive definition, are considered bad requests (note the exclamation point in the regex below). They are dropped by ModSecurity.

```

      :
      :
REQUEST_HEADERS:Host      !^( mail.companyx.com )$      deny
REQUEST_HEADERS:Referer  !^(  .{0,256}                )$      deny
REQUEST_HEADERS:User-Agent !^(  .{0,256}                )$      deny

ARGS:username            !^( [0-9a-zA-Z]{4,16}        )$      deny

ARGS:password            !^(  .{0,16}                 )$      deny
ARGS:submit              !^(  login                   )$      deny
      :
      :
```



Positive Approach – Generated Ruleset

```
<LocationMatch "^/action/submit\.php$">  
  SecRule REQUEST_METHOD "!^(POST)$"  
    "t:none,deny,id:9,status:501,severity:3,msg:'Request method wrong  
    (it is not POST).'"  
  
  ...  
  
  SecRule ARGS_NAMES "!^(username|password|secret)$"  
    "t:none,deny,id:9,status:501,severity:3,msg:'Strict parametercheck:  
    At least one parameter is not predefined for this path.'"  
  
  ...  
  
  SecRule ARGS:username "!^([0-9a-zA-Z]{4,16})$"  
    "t:none,deny,id:9,status:501,severity:3,msg:'Postparameter username  
    failed validity check. Value domain: Custom.'"
```

- Fairly easy to read, no advanced Regex-FU



Entering REMO

- How to get a decent positive Ruleset?
- I need something *bash* won't do! A real editor, a graphical user interface! Something modern! Why must this happen to me? I'm interested in the middle ages!
- **R**ule **E**ditor for **M**Odsecurity as a Ruby on Rails application



<fill in meaningful title here>

- **REMO** core functionality is done
- Not over though, it is beta software
- But it mostly works, it's free and and it's ready to be used by you
- <http://remo.netnea.com>

(this is not an OWASP project. At least not yet)



Securing an SAP application

- SAP applications are getting hooked up to the internet.
- Typical case: E-Recruiting
This means: The whole world is allowed to send requests to the central Human Resources SAP Server and submit crazy attachments on top of that.
- That's like trying to defend a medieval city against infiltration during a public market day.



Remo in action

remo - rule editor for modsecurity

Release 0.2.1-dev

remo
by netnea

clean logfile area import logfile

- view 0: HEAD /heartbeat.html HTTP/1.1 200
- view 1: HEAD /heartbeat.html HTTP/1.1 200
- view 2: GET /sap/bc/bsp/sap/zhrrcf_cand_reg/application.do HTTP/1.1 302
- view 3: GET /sap(bD1lbiZjPTAyNSZkPW1pbG==)/bc/bsp/sap/zhrrcf_cand_reg/application.do HTTP/1.1 200
- view 4: GET /sap/public/bc/ur/design2002/common/Post_E.png HTTP/1.1 200
- view 5: GET /sap(bD1lbiZjPTAyNSZkPW1pbG==)/bc/bsp/sap/zhrrcf_cand_reg/DD22739DAC18EBF1950D001CC4EFE40C HTTP/1.1 200
- view 6: GET /sap/public/bc/ur/design2002/common/Post_E.png HTTP/1.1 200
- view 7: HEAD /heartbeat.html HTTP/1.1 200
- view 8: HEAD /heartbeat.html HTTP/1.1 200
- view 9: HEAD /heartbeat.html HTTP/1.1 200
- view 10: HEAD /heartbeat.html HTTP/1.1 200
- view 11: HEAD /heartbeat.html HTTP/1.1 200
- view 12: HEAD /heartbeat.html HTTP/1.1 200
- view 13: HEAD /heartbeat.html HTTP/1.1 200
- view 14: HEAD /heartbeat.html HTTP/1.1 200
- view 15: HEAD /heartbeat.html HTTP/1.1 200
- view 16: HEAD /heartbeat.html HTTP/1.1 200
- view 17: HEAD /heartbeat.html HTTP/1.1 200
- view 18: HEAD /heartbeat.html HTTP/1.1 200
- view 19: HEAD /heartbeat.html HTTP/1.1 200
- view 20: HEAD /heartbeat.html HTTP/1.1 200
- view 21: HEAD /heartbeat.html HTTP/1.1 200
- view 22: HEAD /heartbeat.html HTTP/1.1 200
- view 23: HEAD /heartbeat.html HTTP/1.1 200
- view 24: HEAD /heartbeat.html HTTP/1.1 200
- view 25: HEAD /heartbeat.html HTTP/1.1 200
- view 26: HEAD /heartbeat.html HTTP/1.1 200

Status: active

new request generate

- HEAD /heartbeat.html
- GET /jobad
 - Remarks: Job AD shortcut
 - Headers
 - Query String Arguments
 - refcode: Letters/Numbers/Space/_/, max. 32 characters mandatory
 - Cookies
 - Post Arguments
- GET /auth/jobs(-int)?.post.ch/.*(html|.css|.png|.jpe?g)
- POST /auth/cgi/jobs(-int)?.post.ch/log(in|out).cgi
- GET (/auth/css/.*|sap/public/.*)(.png|.gif|.css|.js|.htm)
- GET /sap/bc/bsp/sap/hrrcf_start_ext
- GET /sap(\[[a-zA-Z0-9=]{1,32}\])?/bc/bsp/sap/zhrrcf_password/(application.do|1x1)
- POST /sap(\[[a-zA-Z0-9=]{1,32}\])?/bc/bsp/sap/zhrrcf_password/(application.do|1x1)

Successfully toggled mandatory status!



Remo in action

GET /sap/public/bc/ur/design2002/common/Post_E.png

The request **fails** against the present ruleset due to one or multiple parameters.

File, Request number: audit-log-test.log, #4
Status: 200 OK
ModSecurity Message: Warning. Pattern match "HTTP" at REQUEST_PROTOCOL.

Missing mandatory parameters: None. All mandatory parameters present.

Headers

Host: jobs-int.post.ch
User-Agent: Mozilla/5.0 (X11; U; Linux i686; en-US; rv:1.8.1.12) Gecko/20080129 Firefox/2.0.0.10 (Debian-2.0.0.12-0etch1)
Accept: image/png,*/*;q=0.5
Accept-Language: en-us,en;q=0.5
Accept-Encoding: gzip,deflate
Accept-Charset: ISO-8859-1,utf-8;q=0.7,*;q=0.7
Keep-Alive: 300
Connection: keep-alive
Referer: https://jobs-int.post.ch/sap(bD1lbiZJPTAyNSZkPW1pbg==)/bc/bsp/sap/zhrccf_cand_reg/application.do
Cookie: _utma=14378678.39713121.1206026408.1206026408.1206026408.1; RMID=c229d88b47e28070; WT_FF__utmz=14378678.1206026408.1.1.utmccn=(direct)|utmcsr=(direct)|utmcmd=(none); ittrksessid=10.226.back_jobs-int.post.ch=https%3a%2f%2fjobs-int.post.ch%2fsap%2fbc%2fbsp%2fsap%2fhrrcf_start_ext%3f

Cookie Parameters

back_jobs-int.post.ch: https%3a%2f%2fjobs-int.post.ch%2fsap%2fbc%2fbsp%2fsap%2fhrrcf_start_ext%3fsap-language%3dDE%

Querystring Parameters

Post Parameters



Status

- It's beta software and it needs more work.
The list of feature requests is impressive, btw.
- Code quality is ok
(but it could do better security-wise)
- It's really a simple piece of software that facilitates a boring task: Writing ModSecurity rules yourself.
- Did I mention it needs more work?
(This is where you come in)



Final Words

- Johann Peeters' talk about input validation
Be really careful to set up your process properly or you enter maintenance hell.
- Combining Remo and Core Rules? Yes! Negative first, positive afterwards.

<http://remo.netnea.com>

