Hands-On

ModSecurity and Logging

Philipp Krenn

@xeraa

Let's talk about security...


THIS IS FINE.

A1:2017-Injection

https://www.owasp.org/index.php/Top_10-2017_Top_10


THIS IS FINE.

A10:2017-Insufficient Logging & Monitoring

https://www.owasp.org/index.php/Top_10-2017_Top_10


TH-
THIS IS NOT FINE!!
OH MY GOD EVERY THINGS ON FIRE
WHAT THE HELL IS MY PROBLEM

elastic

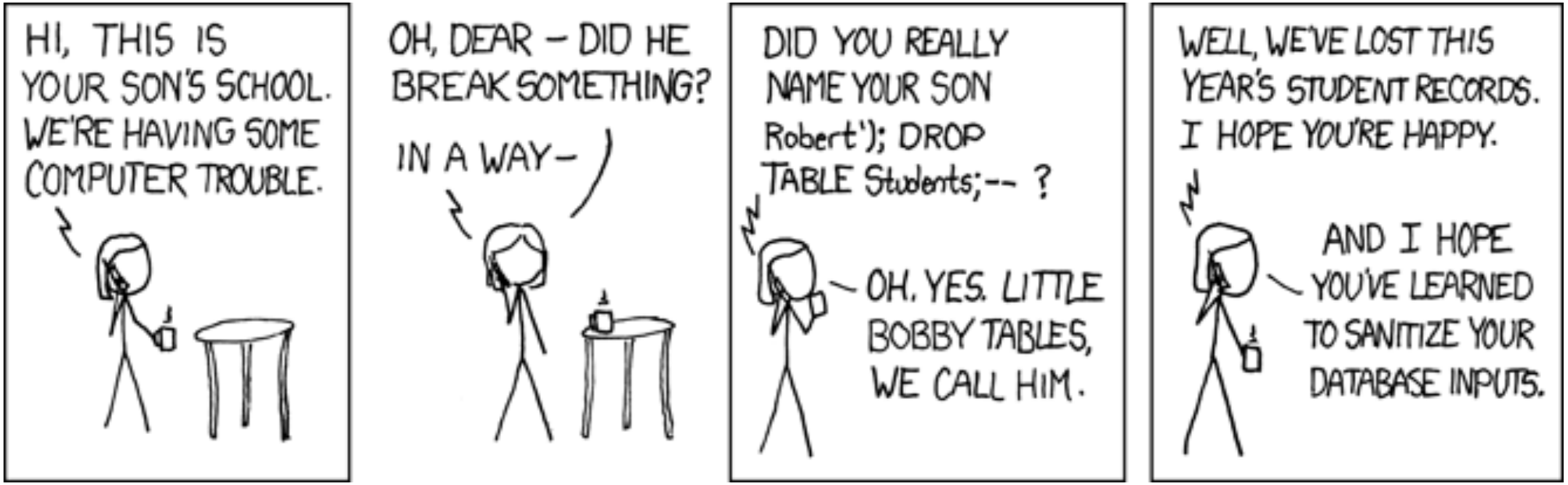
Developer 🥑

Disclaimer

I build **highly** monitored Hello World apps

Hello World of SQL Injection:
https://xeraa.wtf

https://xeraa.wtf/read.php?id=1 🤔


HI, THIS IS YOUR SON'S SCHOOL. WE'RE HAVING SOME COMPUTER TROUBLE.
OH, DEAR – DID HE BREAK SOMETHING?
IN A WAY–
DID YOU REALLY NAME YOUR SON Robert'); DROP TABLE Students;-- ?
OH. YES. LITTLE BOBBY TABLES, WE CALL HIM.
WELL, WE'VE LOST THIS YEAR'S STUDENT RECORDS. I HOPE YOU'RE HAPPY.
AND I HOPE YOU'VE LEARNED TO SANITIZE YOUR DATABASE INPUTS.


```
python sqlmap.py --url "https://xeraa.wtf/read.php?id=1" --purge
```

Injection

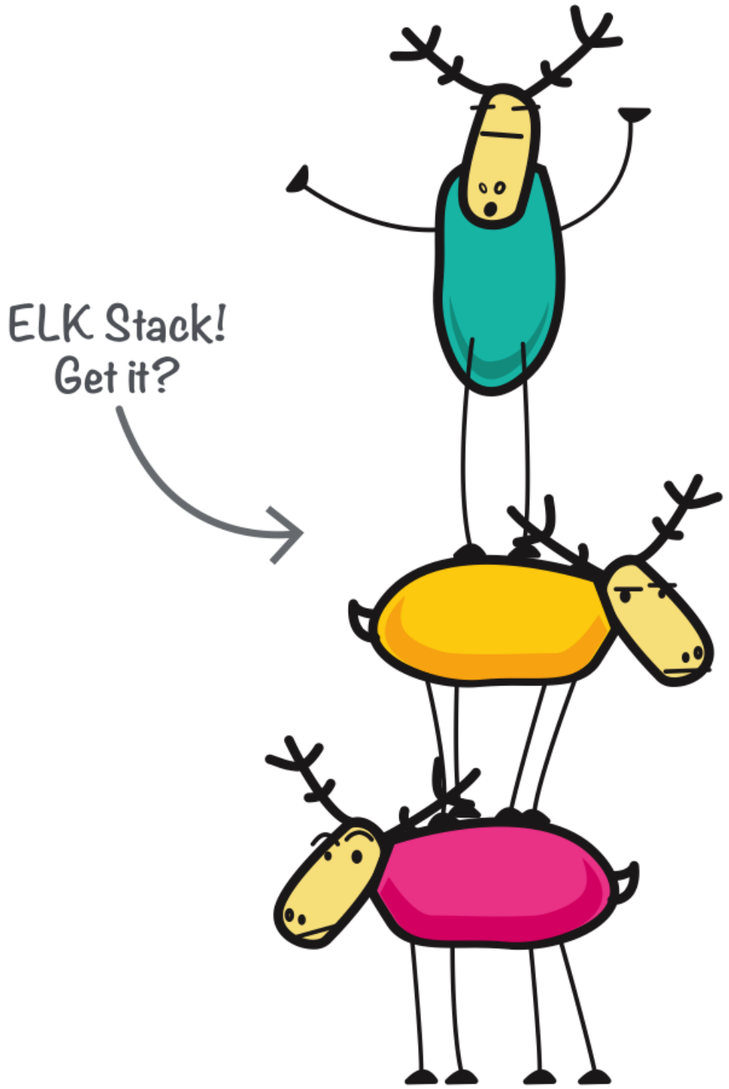
```
;INSERT INTO employees (id,name,city,salary) VALUES
(4,'new','employee',10000)
```

No Escaping Either

```
;INSERT INTO employees (id,name,city,salary) VALUES
  (5,'<script>alert("hello")</script>','evil',0)
```


ALL THE THINGS!


me looking
for the bug
7.2 GB
of log file



E Elasticsearch

L Logstash

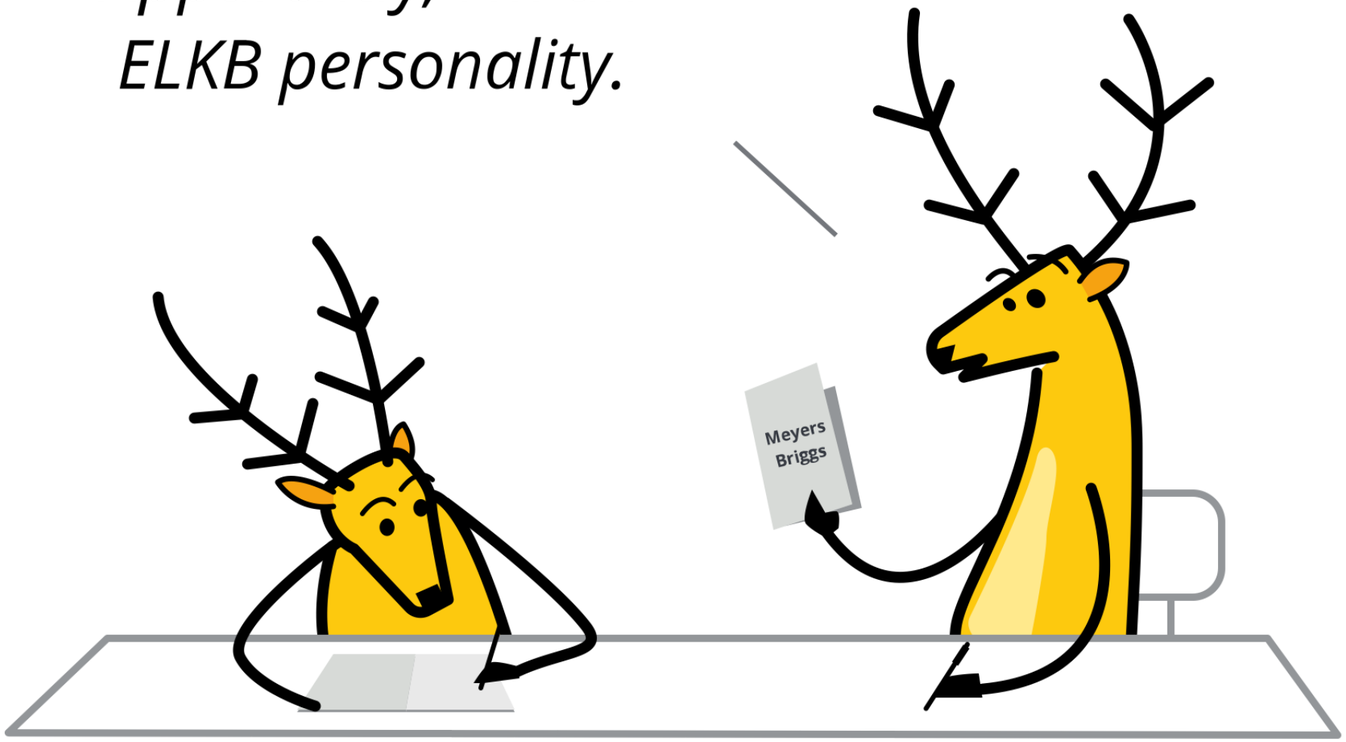
K Kibana

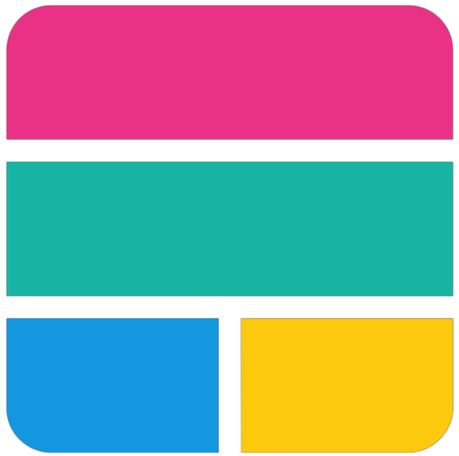

lyft


slack

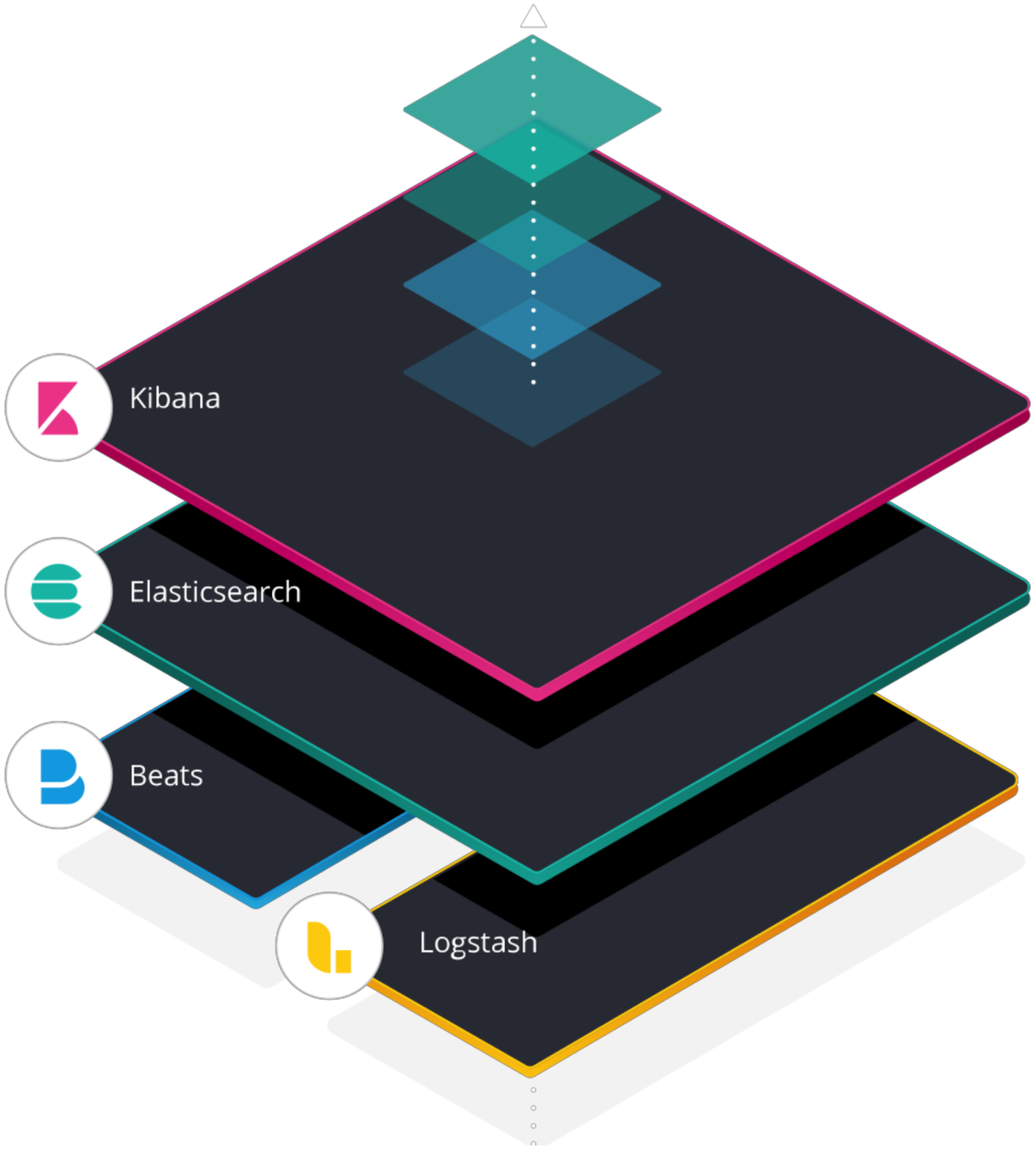

fitbit

Apparently, I'm an ELKB personality.





elastic stack


Kibana
Elasticsearch
Beats
Logstash

What's going on in our app?

55737a,0x71717a6271),NULL,NULL,NULL-- YJoN

2019-04-05 12:07:03.679 SQL query: SELECT * FROM employees WHERE id = 1 UNION ALL SELECT NULL,NULL,NULL,CONCAT(0x717a766a71,0x516950484c527a677758,0x71717a6271)-- qFgZ

2019-04-05 12:07:03.679 SQL query: SELECT * FROM employees WHERE id = 1 UNION ALL SELECT NULL,CONCAT(0x717a766a71,0x744f5352425953674669,0x71717a6271),NULL,NULL-- SBru

2019-04-05 12:07:03.679 SQL query: SELECT * FROM employees WHERE id = -6655 UNION ALL SELECT CONCAT(0x717a766a71,0x664d6a6268664b41494f637a6575785576415741424755 4f5552745544584d676c576c4152795165,0x71717a6271),NULL,NULL,NULL-- YiVs

2019-04-05 12:07:04.679 SQL query: SELECT * FROM employees WHERE id = -6770 UNION ALL SELECT NULL,NULL,NULL,CONCAT(0x717a766a71,0x774c6653426f53436a567557617369556a4b416e516757576c465a526e484a717373566b7359726e,0x71717a6271)-- jdLc

2019-04-05 12:07:04.679 SQL query: SELECT * FROM employees WHERE id = -7077 UNION ALL SELECT NULL,NULL,NULL,CONCAT(0x717a766a71,(CASE WHEN (1022=
1022) THEN 1 ELSE 0 END),0x71717a6271)-- qIrV

2019-04-05 12:07:05.680 SQL query: SELECT * FROM employees WHERE id = ''

2019-04-05 12:07:08.680 SQL query: SELECT * FROM employees WHERE id = ''

2019-04-05 12:08:13.684 SQL query: SELECT * FROM employees WHERE id = ''select * from contacts

2019-04-05 12:08:38.686 SQL query: SELECT * FROM employees WHERE id = ''select * from contacts

2019-04-05 12:09:33.690 SQL query: SELECT * FROM employees WHERE id = ;select * from contacts

2019-04-05 12:09:40.691 SQL query: SELECT * FROM employees WHERE id = 1;INSERT INTO employees (id,name,city,salary) VALUES (4,'test','test',10000)

2019-04-05 12:09:55.693 SQL query: SELECT * FROM employees WHERE id = 4

2019-04-05 12:09:55.693 SQL query: SELECT * FROM employees WHERE id = ;select * from contacts

2019-04-05 12:12:30.698 SQL query: SELECT * FROM employees WHERE id = 3

2019-04-05 12:12:55.700 SQL query: SELECT * FROM employees WHERE id = 3;drop table employees

2019-04-05 12:13:02.701 SQL query: SELECT * FROM employees WHERE id = 4

2019-04-05 12:13:27.703 SQL query: SELECT * FROM employees WHERE id = 4;drop table employee

DELETE or DROP?

OWASP ModSecurity Core Rule Set

THE 1ST LINE OF DEFENSE

elastic

@xeraa

ModSecurity is an open source, cross-platform web application firewall (WAF) module. Known as the "Swiss Army Knife" of WAFs, it enables web application defenders to gain visibility into HTTP(S) traffic and provides a power rules language and API to implement advanced protections.

OWASP ModSecurity Core Rule Set (CRS) Version 3

- HTTP Protocol Protection
- Real-time Blacklist Lookups
- HTTP Denial of Service Protections
- Generic Web Attack Protection
- Error Detection and Hiding

Commercial Rules from Trustwave SpiderLabs

- Virtual Patching
- IP Reputation
- Web-based Malware Detection
- Webshell / Backdoor Detection
- Botnet Attack Detection
- HTTP Denial of Service (DoS) Attack Detection

Run sqlmap again

```
python sqlmap.py --url "https://xeraa.wtf/read.php:8080?
id=1" --purge
```

Custom Rule

```
SecRule REQUEST_FILENAME "form.php" "id:'400001',chain,deny,log,msg:'Spam detected'"
SecRule REQUEST_METHOD "POST" chain
SecRule REQUEST_BODY "@rx (?i:(pills|insurance|rolex))"
```


WHAT THE FUCK WAS
I EVEN THINKING

Conclusion

Examples

https://github.com/xeraa/mod_security-log

Code

Logging

ModSecurity

Questions?

Philipp Krenn @xeraa