Bypassing malware detection mechanisms in online banking

Jakub Kałużny

Mateusz Olejarka

Who are we?

- Pentesters @ SecuRing
- Ex-developers
- Experience with:
  - E-banking and mobile banking systems
  - Multi-factor and voice recognition authentication
  - Malware post mortem

@j_kaluzny

@molejarka

securing

Agenda

- Intro
  - Why this topic?
  - How it’s done?
  - Will it blend?
- Vulnerabilities
- Conclusions
- Q&A*

Intro

Why this topic ?

- AVs are not reliable
- Users are lazy
- Market gap for new solutions
- A lot of money

How malware works?

- Interaction with browser
  - Web injects
  - Other?
- What it does
  - Steals credentials
  - Changes transaction data
  - Automates attacks

zeus

torpig

citadel

carberp

gozi

spyeye

bugat

zitmo

eblaster

vbclip

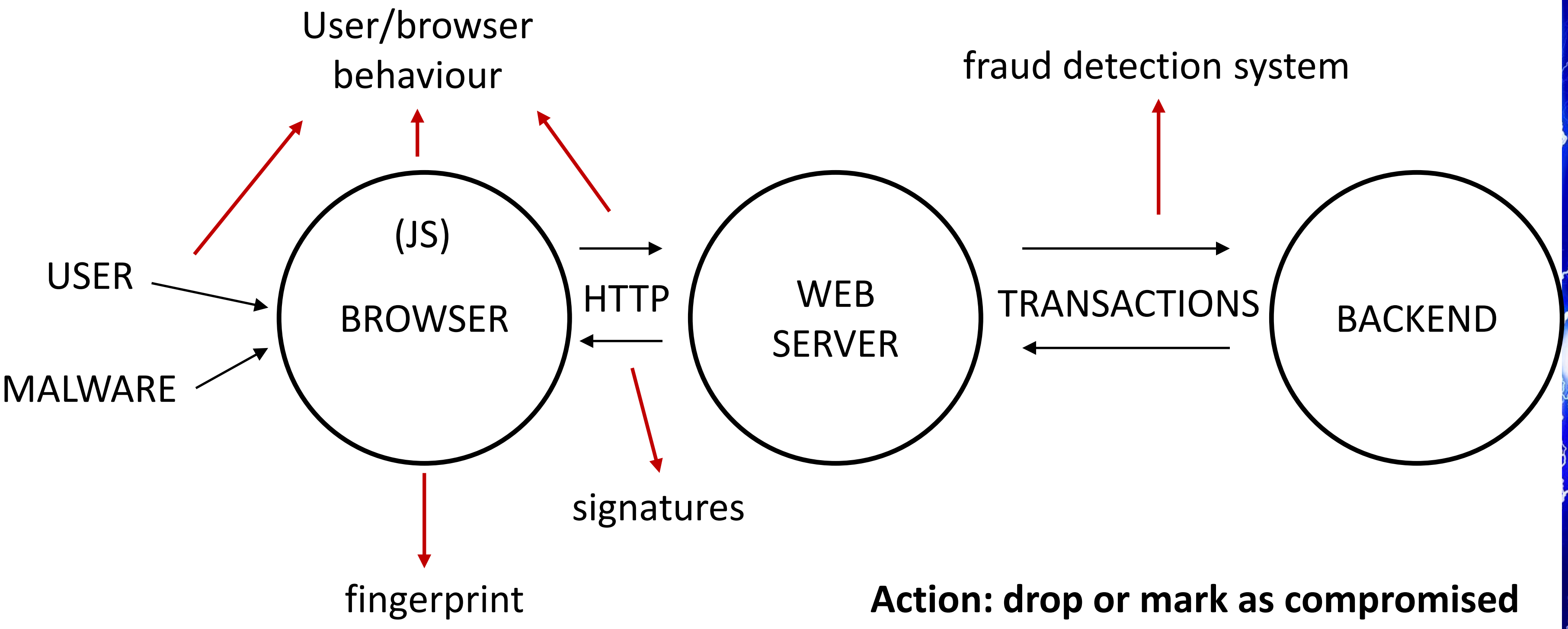
banatrix

carbanak

hiloti

What is online malware detection ?

Aim: Detect malware presence

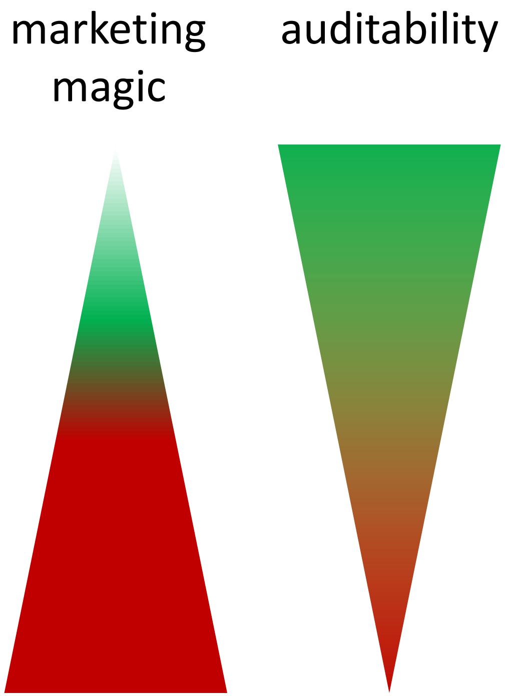


Action: drop or mark as compromised

What are the limits ?

Malware detection methods:

- HTTP response signature
- Browser fingerprint
- User/browser behavior
- Server-side behavioral methods
- Fraud detection system



marketing magic

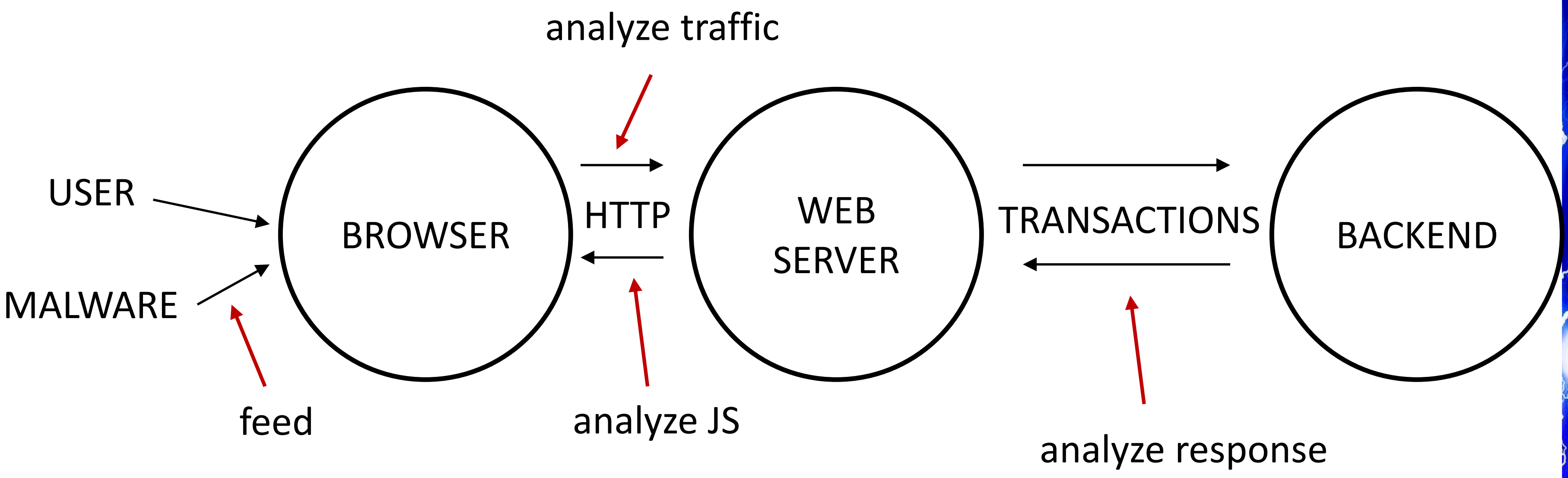
auditability

What is the purpose of this report?

- We do not represent any vendor
- We want to show
  - — architecture failures
  - — implementation errors
- We want to talk about what can be done

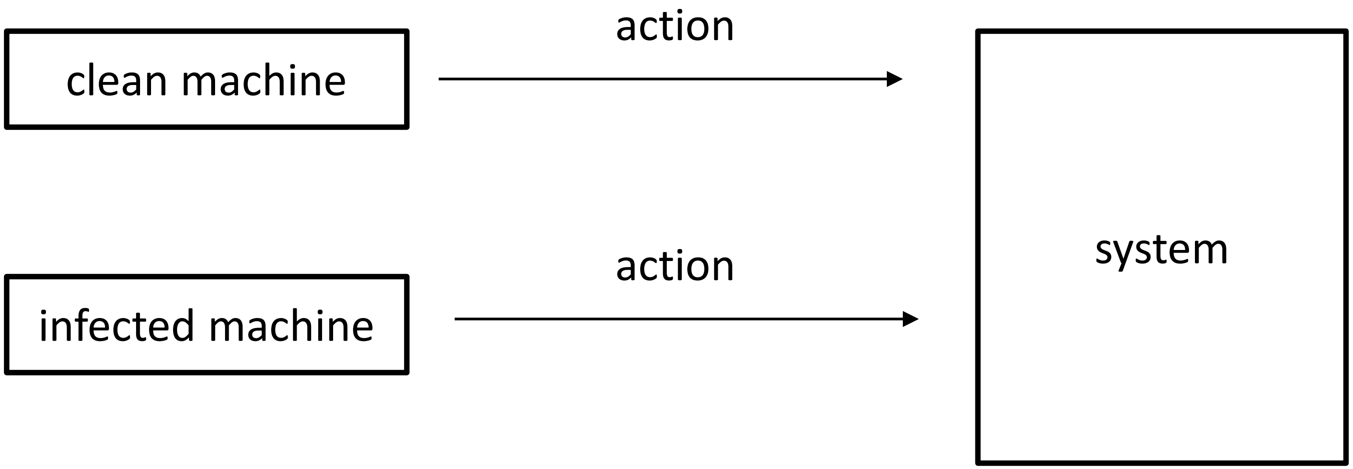
Vulnerabilities

Our approach



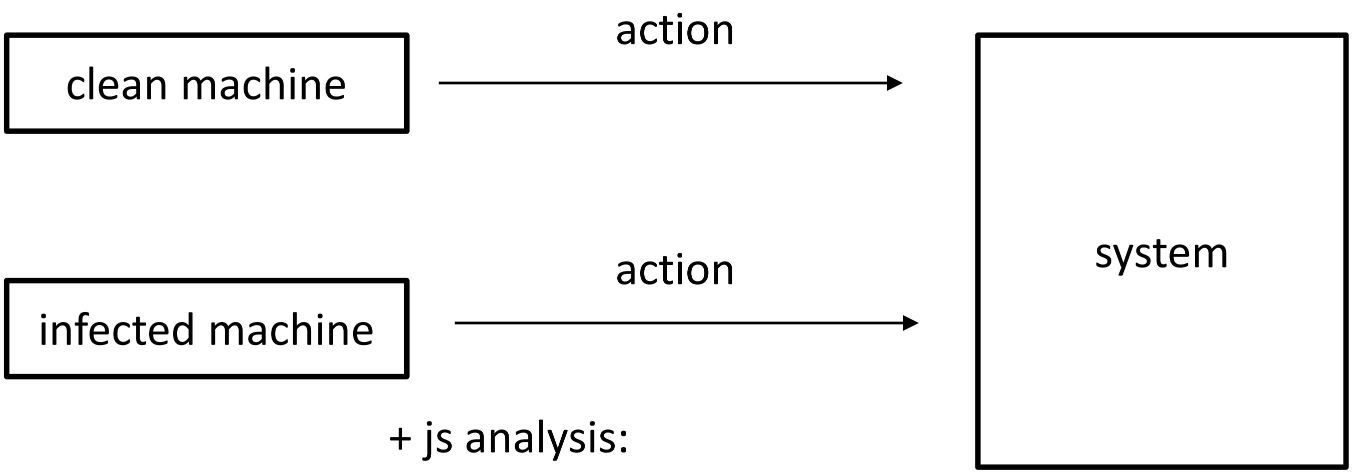
First idea

HTTP traffic



Going through…

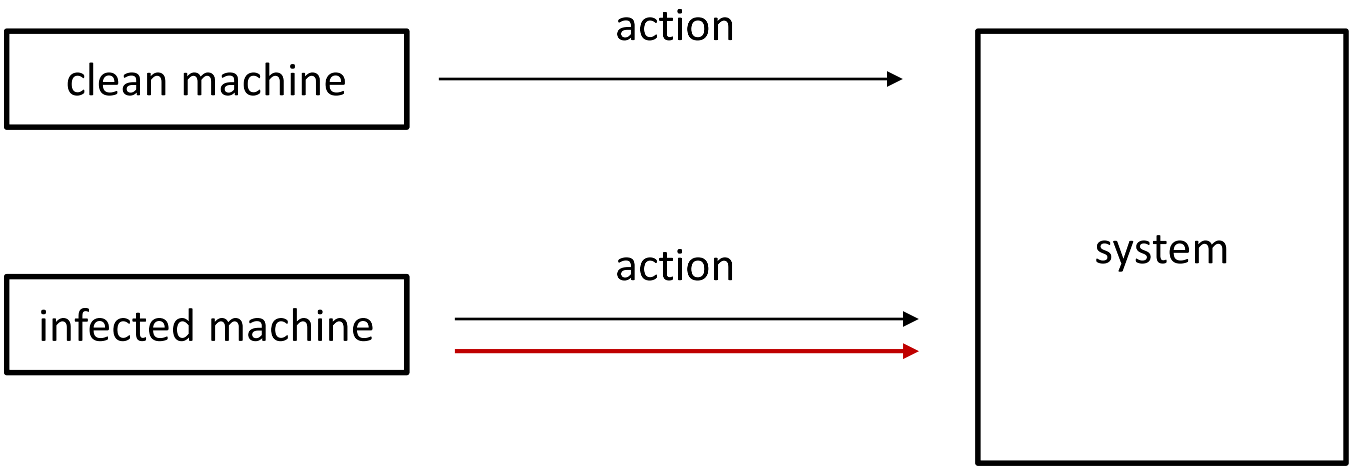
HTTP traffic + JS analysis



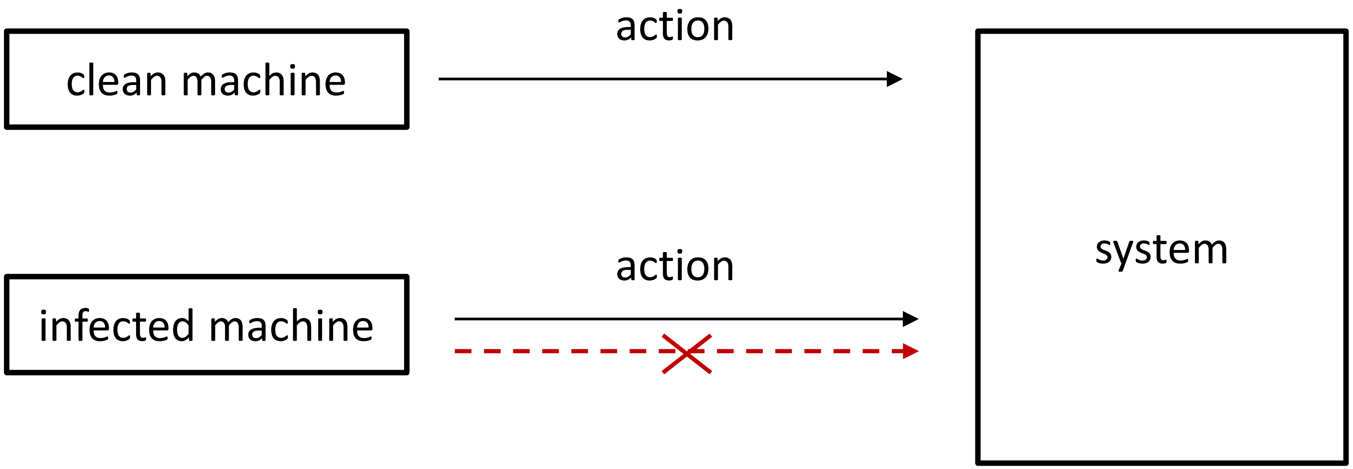
+ js analysis:

- Different paths
- Different subdomains
- Different data format (e.g. base64)
- Encryption (e.g. rsa)

Almost there...

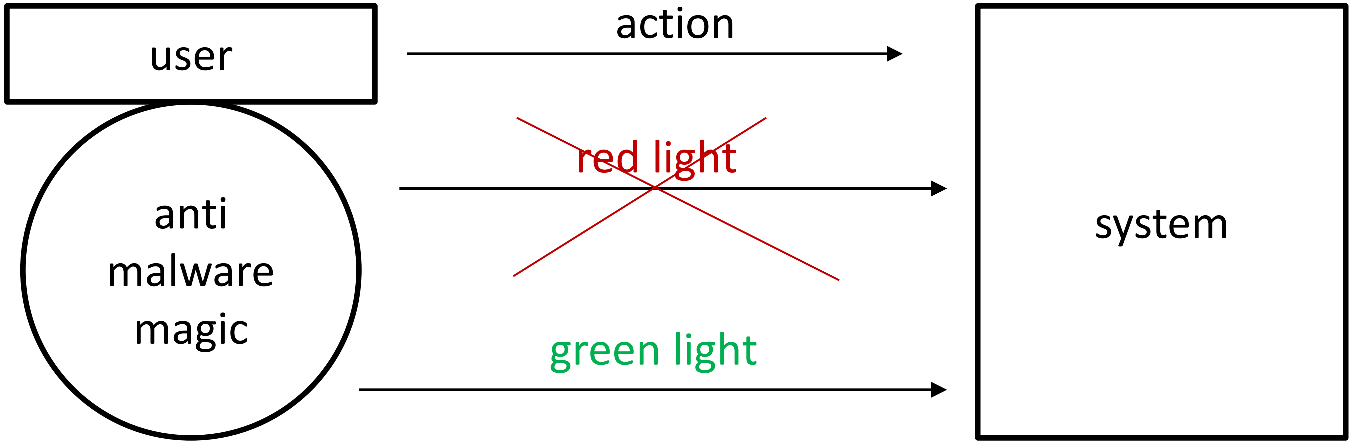


If it bleeds, we can kill it



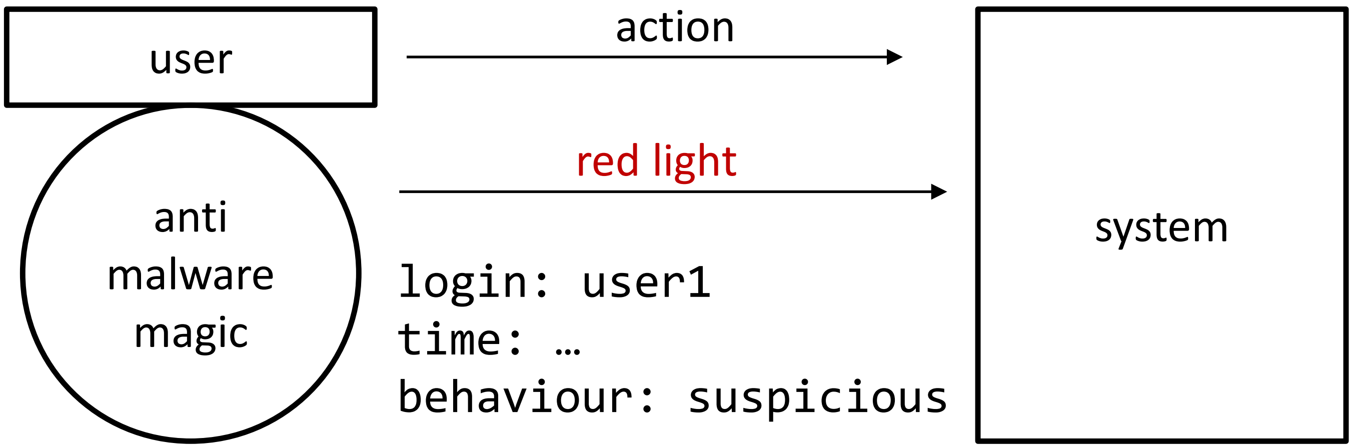
BYPASSED!

Architecture problem



Words of wisdom: adverse inference

Malware spotted!



user

action

red light

anti malware magic

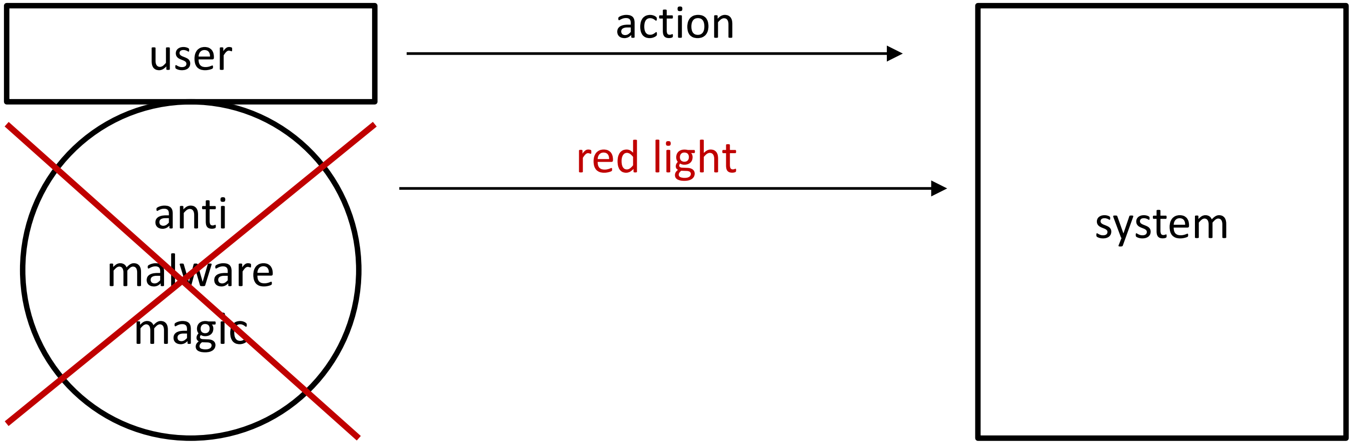
system

```
login: user1
time: …
behaviour: suspicious
```

```
login: user2?
```

Who sends the alert ?

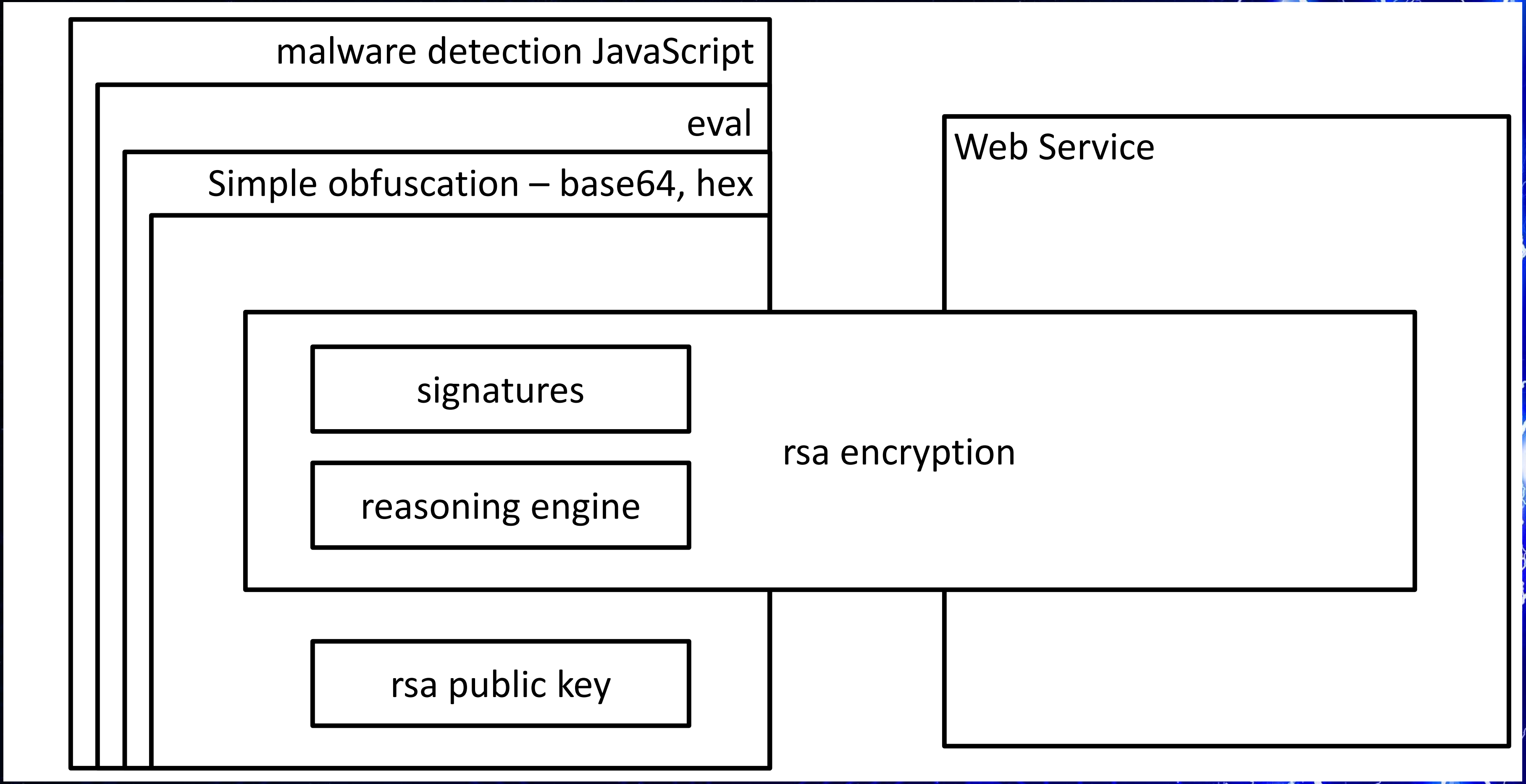
First things first



JavaScript
slowing your page ?

BYPASSED!

Security by obscurity



malware detection JavaScript

eval

Simple obfuscation – base64, hex

signatures

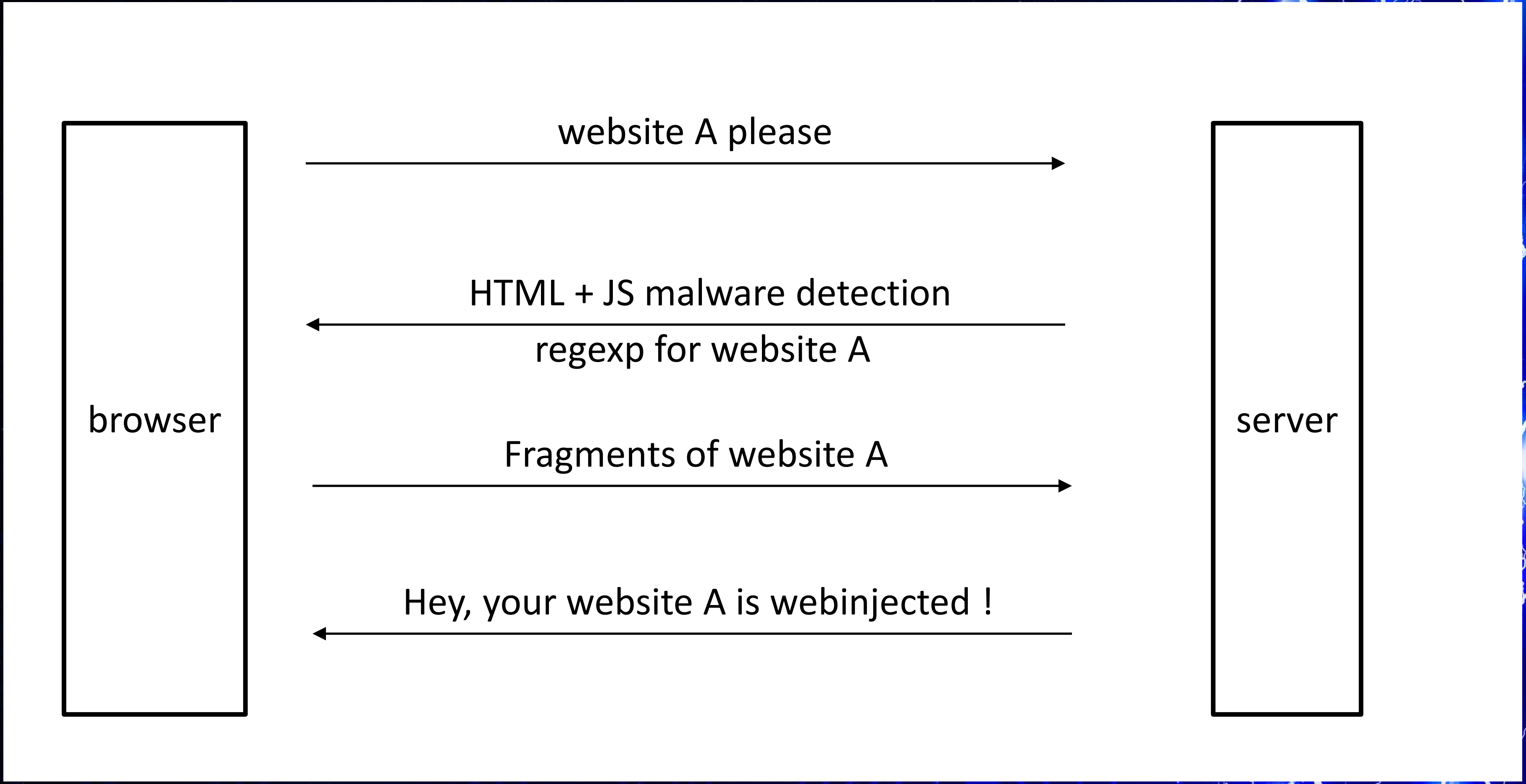
reasoning engine

rsa public key

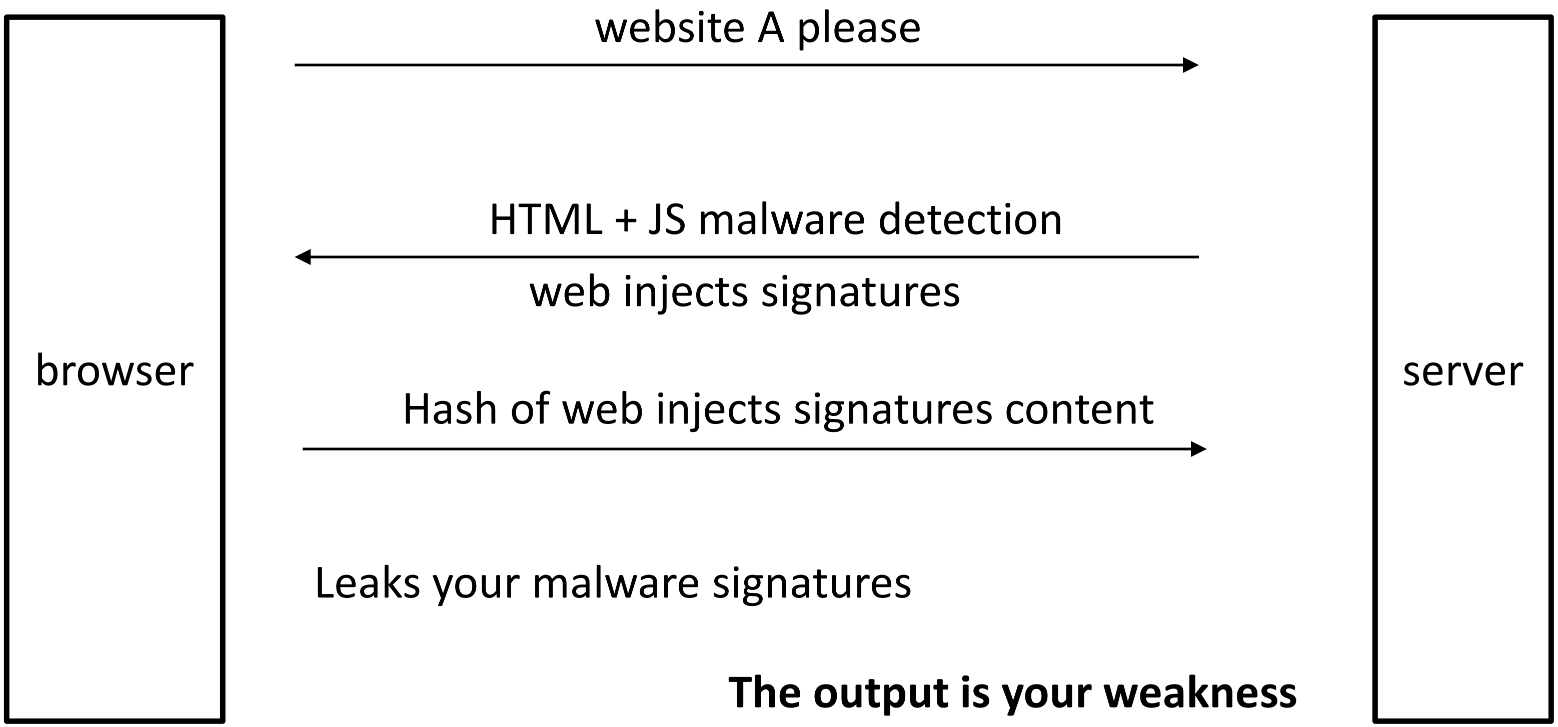
rsa encryption

Web Service

Signatures server-side



Signatures client-side



browser
server
website A please
HTML + JS malware detection
web injects signatures
Hash of web injects signatures content

Leaks your malware signatures

The output is your weakness

Conclusions

Conclusions - banks

- Buy an anti-malware box?
- Better call your crew
- Trust, but verify
- Ask for technical details

Conclusions – vendors

- Online malware detection is a good path, behavioral systems are a future of ITsec
- But they are still based on the old HTTP + HTML + JS stack
- Think about architecture and implementation

What's next?

- Recommendations for potential anti-malware buyers – paper, work in progress
- Interested? -> malware@securing.pl or antimalware@securing.pl

Thank You

Q&A*