JavaScript Agent Injection

Ofer Shezaf,
ofers@breach.com

CTO, Breach Security
OWASP Israel leader
ModSecurity Core Rule Set Leader

Introduction

Ofer Shezaf

- Chief Technology Officer at Breach Security.
- In charge of application security research.
- Based out of Tel-Aviv, Israel.
- Background in National Information Security.
- Open Source and Community projects:
  - Board Member, Web Application Security Consortium.
  - President, OWASP Israeli chapter.
  - Project Leader, ModSecurity Core Rule Set Project.
  - Project Leader, WASC Web Hacking Incident Database.

BREACH

Breach Security

Technology Leaders

- Headquarters in Carlsbad, CA, with R&D Center in Herzliya, Israel and London, UK.
- Experience with Web security solutions since 1999.
- Managed by an experienced group of security professionals.
- Best application security DNA in the industry. We wrote the book.

BREACH

BREACH

Shift in Paradigm

State of Defense Against the Dark Arts

Perimeter Defense:

- Heavily guarded border separating between our territory and the enemy's.
- Effective against high profile low impact hostile activities such as terrorism.

BREACH

State of Defense Against the Dark Arts

Security Layers:

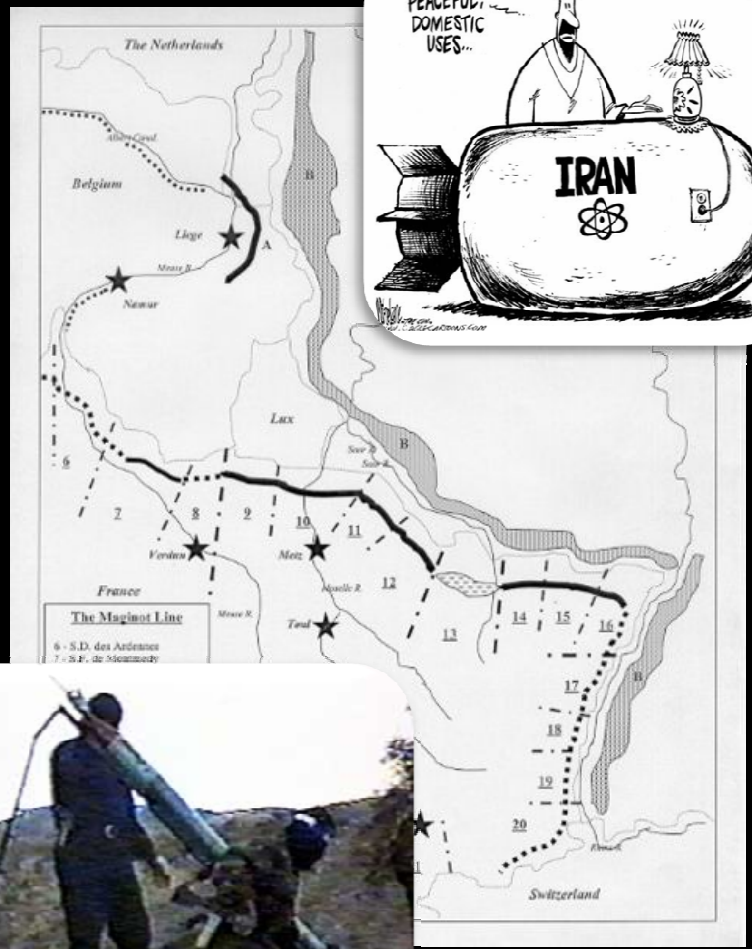
- Internal defense lines for backup if perimeter is bypassed.
- Also aimed at inside security threats.

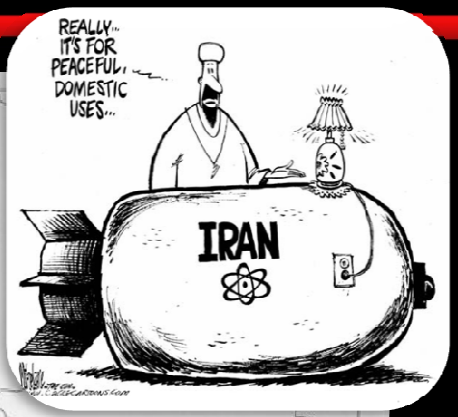
BREACH

The Maginot Line

Lessons learned from the battle field:

- Attack is the best form of defense!
- Defense has to be dynamic: static patterns are bound to be taken advantage of.
- Any barrier can be bypassed.





BREACH

Let's be proactive

Move the battle ground out of the data center.

Ground rules:

- Follow net ethics, go only where invited.
- Be unpredictable, it is enemy land and you will caught if static.
- We reversed roles, now we are the virus evading detection.

BREACH

BREACH

JavaScript Agents Injection

Content Injection

Use JavaScript Code for security on the client:

- JavaScript attack footprint is minor on the server and huge on the client.
- Protection from client only attacks:
  - DOM-based Cross-Site Scripting.
  - Universal PDF XSS.
- Protection from trust zones violation attacks:
  - Protection from an application on the same host (Amit Klein, “Path Insecurity”).

Inject it using a proxy or a WAF:

- Ensure complex, dynamic behaviour independent of the application, including obfuscation & polymorphism.
- Continues updates and potentially even an “in the cloud” service.
- Provide protection for non-HTML pages by wrapping them in HTML (redirect, refresh, frames).

BREACH

And the Agent's Payload?

Secure session management

- Avoid the evasion options of Cookies & HTTP authentication (Amit Klein, "Path Insecurity")
- One time tokens "super" digest authentication.

Intercept JavaScript Events to perform:

- Client side input validation (Amit Klein, "DOM Based Cross Site Scripting or XSS of the Third Kind")
- DOM Hardening
- Rules/Signature based negative security.

BREACH

Caveats

- Like any agent in hostile territory, ours is also exposed and vulnerable.
- Where is the border line between protecting the server and providing general client security?
  - Wouldn't an ActiveX be more effective?
  - And consequently, what is the business model?

BREACH

Content Injection in ModSecurity

ModSecurity (2.5+) supports content injection:

- Prepend & Append to HTML.
- Injection at arbitrary location will be added in the future.

Example code:

```
SecRule RESPONSE_CONTENT_TYPE ^text/html \
"phase:3,nolog,pass,prepend:'PAGE_HEADER<hr>'"
```

With JavaScript:

```
SecRule RESPONSE_CONTENT_TYPE ^text/html \
"phase:3,nolog,pass,prepend:\
'<script>document.write(\'Hello World\')</script>'
```

- .

BREACH

Credits

Amit Klein: "DOM Based Cross Site Scripting or XSS of the Third Kind"

Ivan Ristic: "Protecting Web Applications from Universal PDF XSS"

Thank You!

Ofer Shezaf
ofers@breach.com

BREACH