

mod_antimalware: a novel web server module for containing web-based malware infections

Neil Daswani

www.neildaswani.com

CTO & Co-Founder

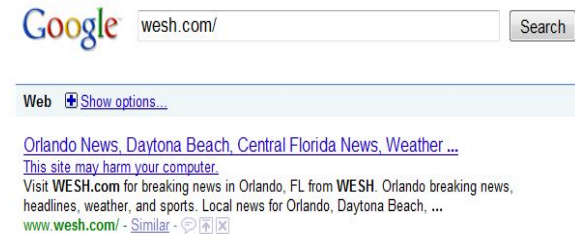
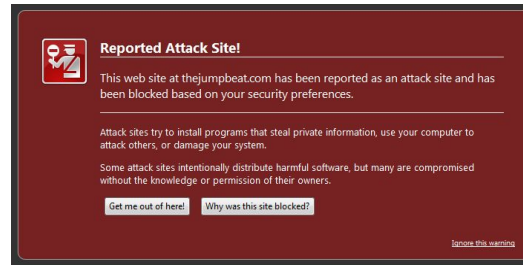
Joint work with Pete Fritchman, Ameet Ranadive, Shariq Rizvi, Ravi Reddy

www.dasient.com

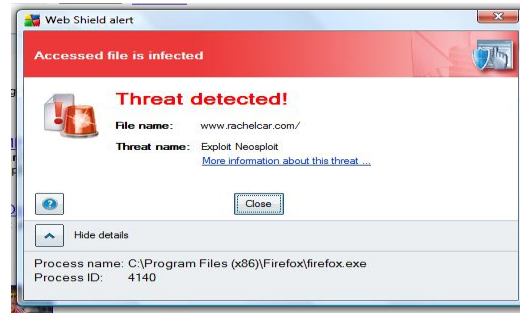


Web Malware Attacks Hurt Enterprises

Traffic and
revenue loss



Brand and
customer loss



Data Theft/
Compliance
Liability



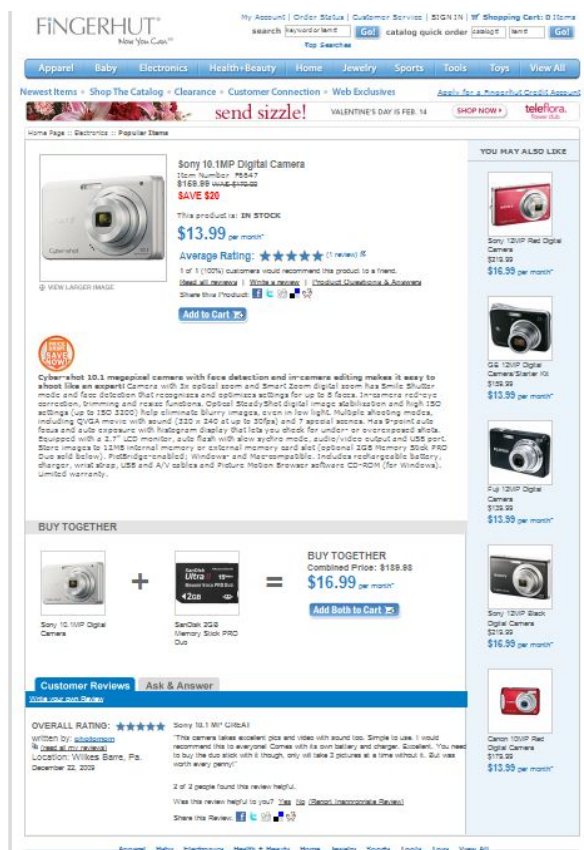
The Challenge for Websites: Many Ways to Get Infected

Web 2.0/ external content

- Mash-ups
- Widgets
- External images
- User generated content (HTML, images, links, exe, documents)
- Third-party ads

Passwords compromised

- FTP credentials
- SSH credentials
- Web server credentials



Software vulnerabilities

- SQL injection
- XSS
- PHP file include
- Unpatched Software (blog, CMS, shopping cart, web server, PHP, Perl)

Infrastructure vulnerabilities

- Vulnerable hosting platform
- Network vulnerabilities

“Structural” Vulnerabilities

A structural vulnerability is:

a weakness in a web page that may allow an attacker to compromise the entire page as a result of the reliance of the page design on a page component, where the compromise of the component can result in compromise of the entire page.

Examples

```
<script> ... document.write(unescape("%3Cscript  
src="" + gaJsHost + "google-analytics.com/ga.js" ...  
</script>
```

```
<script type='text/javascript' src='jquery.js' ...
```

```
<iframe src="http://ad.yieldmanger.com/iframe?...  
%26search%3D%5Bterms%5D%26ip%3D%5Bip  
%5D%26ua%3D%5Bua%5D%26style  
%3D2%26size%3D160x600,Z%3D160x600...
```

Structural Vulnerabilities: Stats

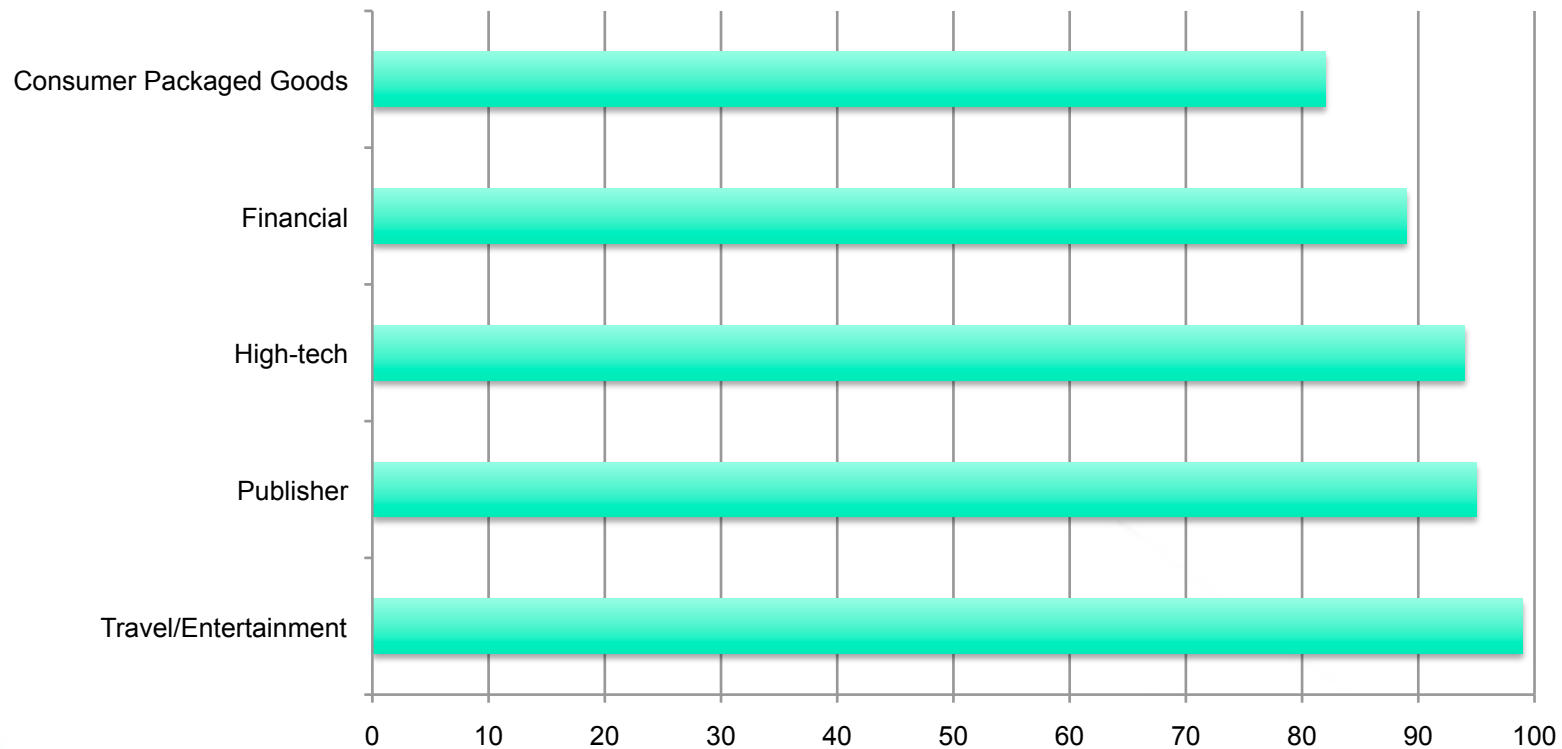
75% of web sites use third-party JavaScript widgets (analytics, UI, ads, etc)

82% of publishers run third-party ads

91% of businesses have some outdated (vulnerable) third-party software powering their websites

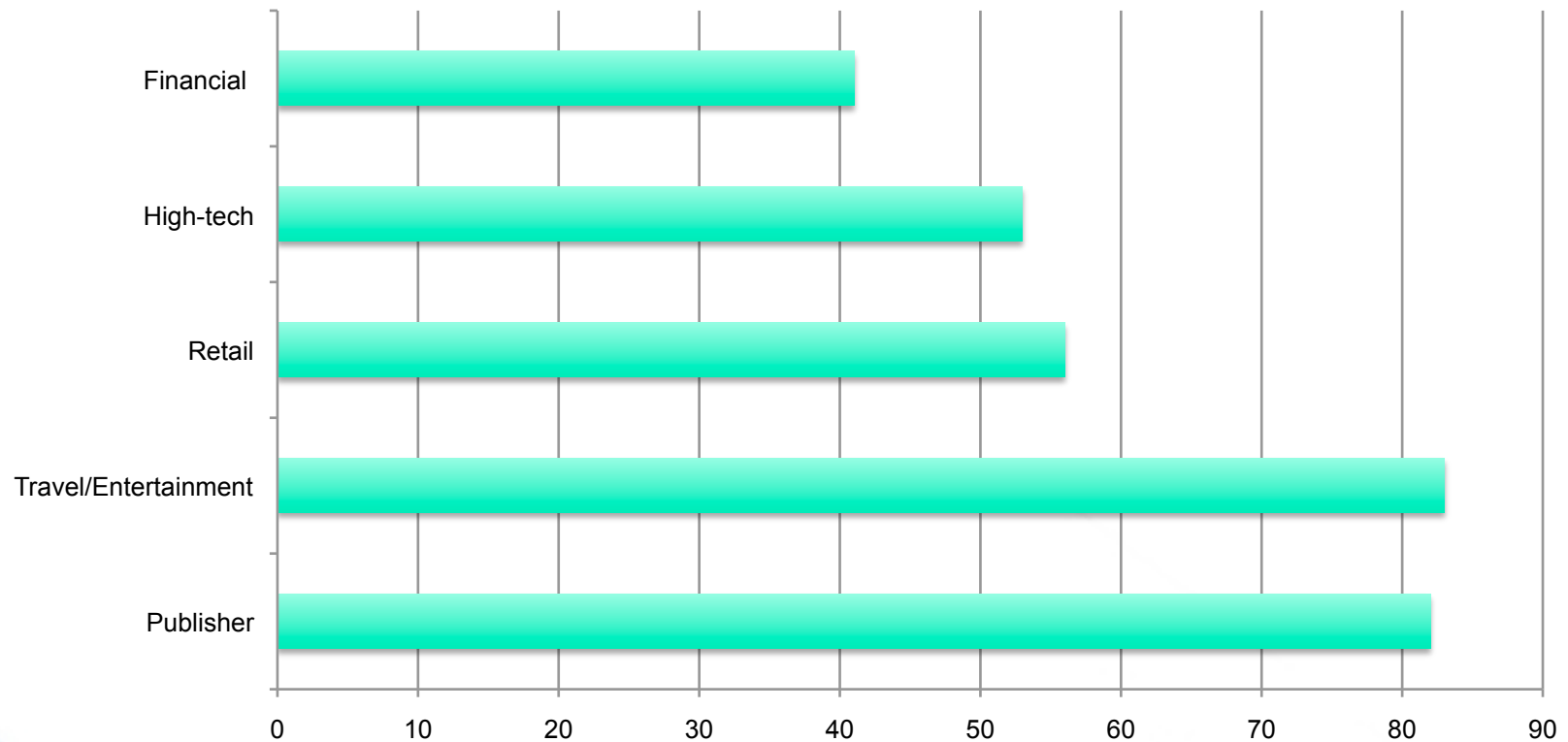
Structural Vulnerabilities: Stats

3rd-Party JavaScript Usage (by vertical)



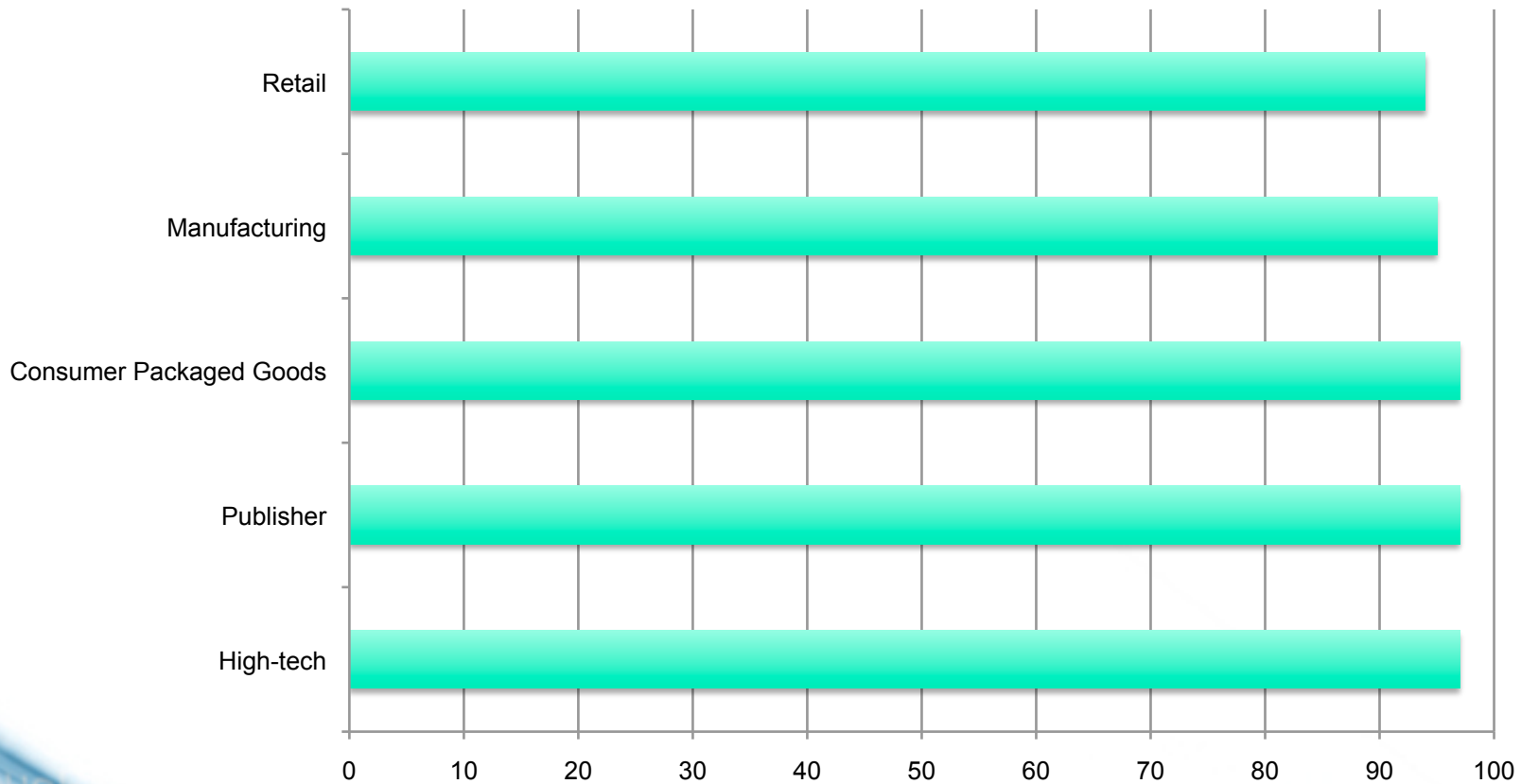
Structural Vulnerabilities: Stats

3rd-Party Advertising

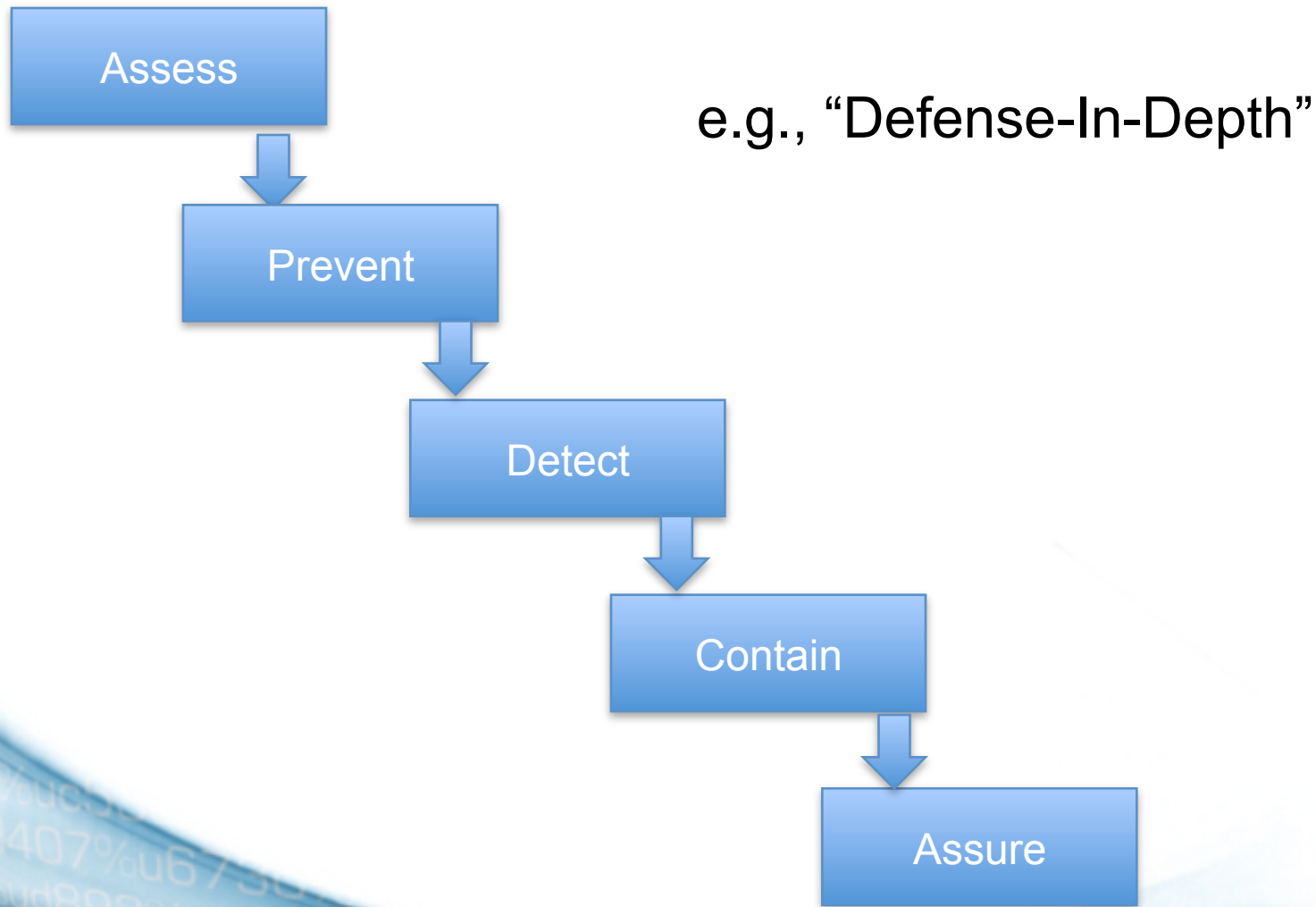


Structural Vulnerabilities: Stats

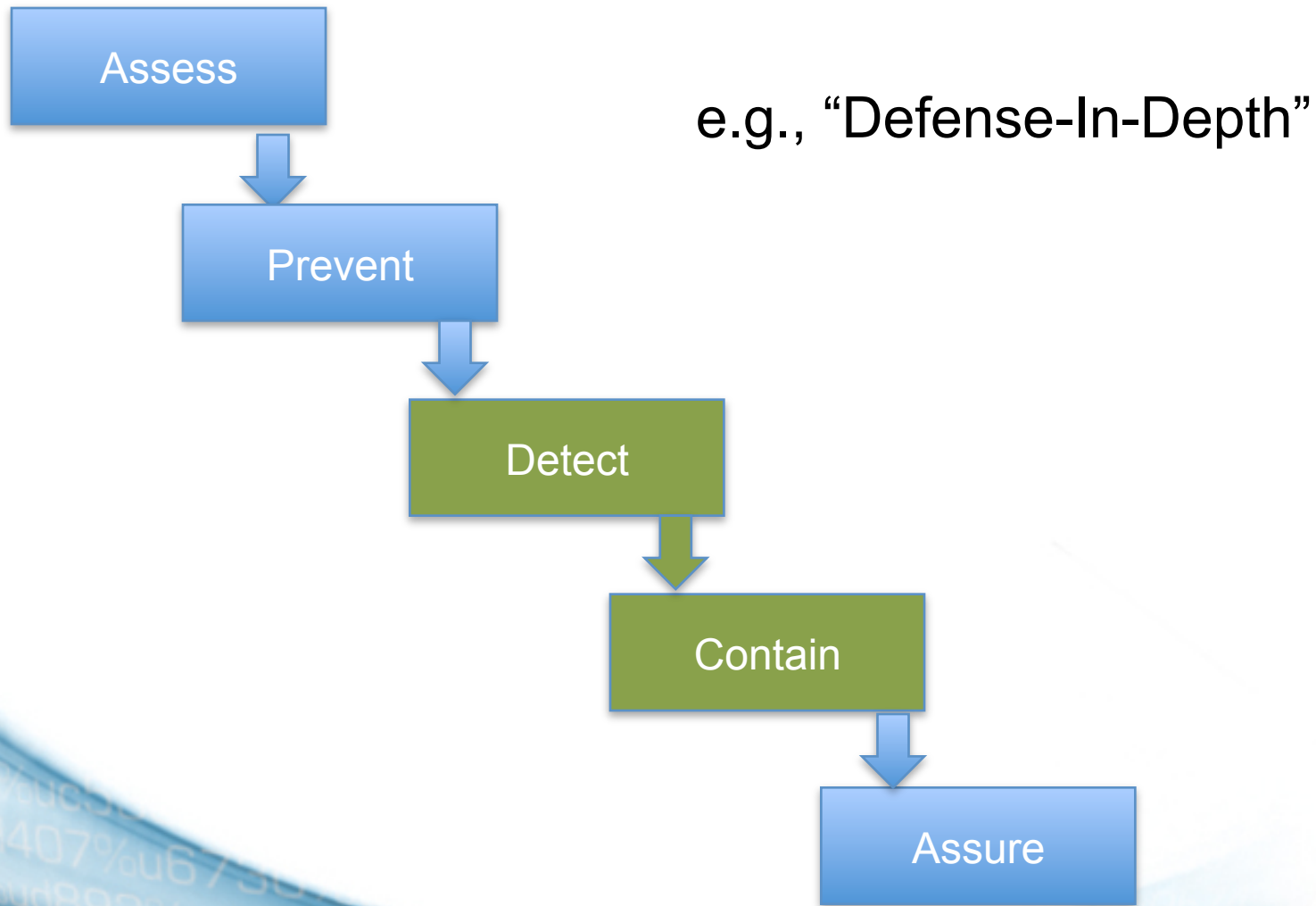
Outdated Web Apps / Packages



Problem: How to Provides the Complete Lifecycle of Malware Protection for Web Sites?



Problem: How to Provides the Complete Lifecycle of Malware Protection for Web Sites?



Why is protecting web sites from drive-bys hard?

Need to bring “lifecycle” of protection to the web

Need to “root cause” what code on the page caused the problem

Need to be able to parse page in real time and strip out infection. (Could be coming from anywhere—file, DB, etc)

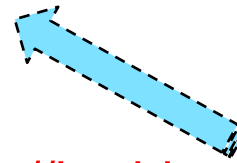
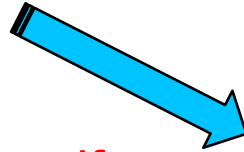
Need to do so with high performance



Solution: Detection & Containment

- Goal: Extract “root cause” of malcode

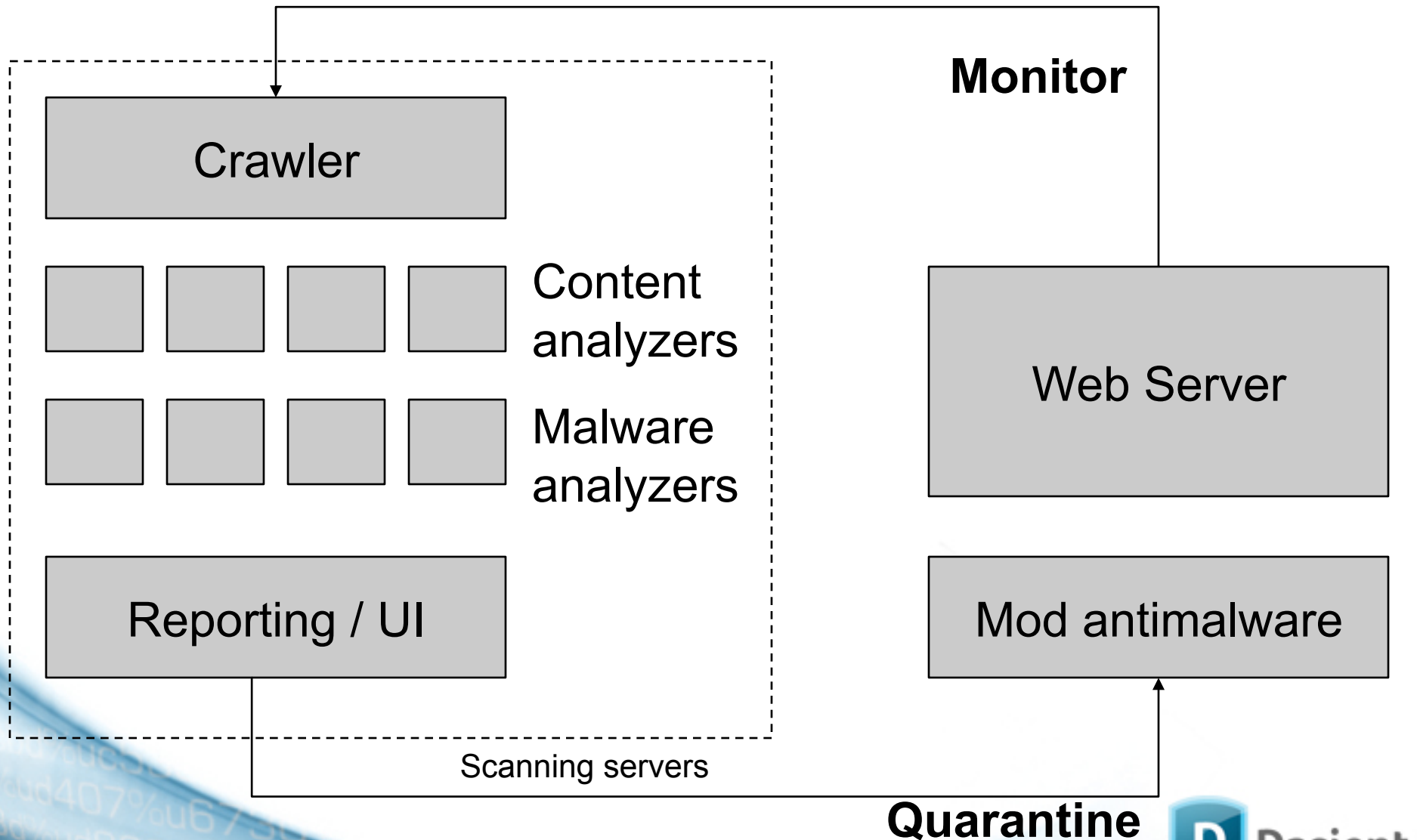
<script src="http://external.com/a.js">



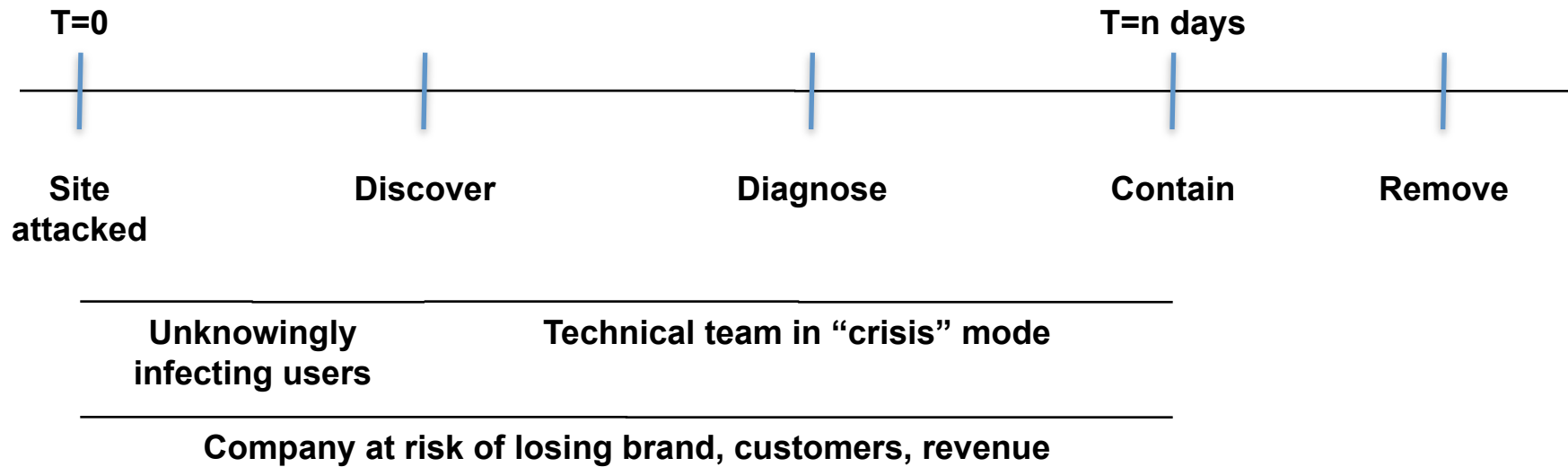
<iframe src="http://baddomain.com">

- Detection
 - Behavioral Content Extraction (active scripts)
 - Lineage computation
 - Features / Signals Analysis

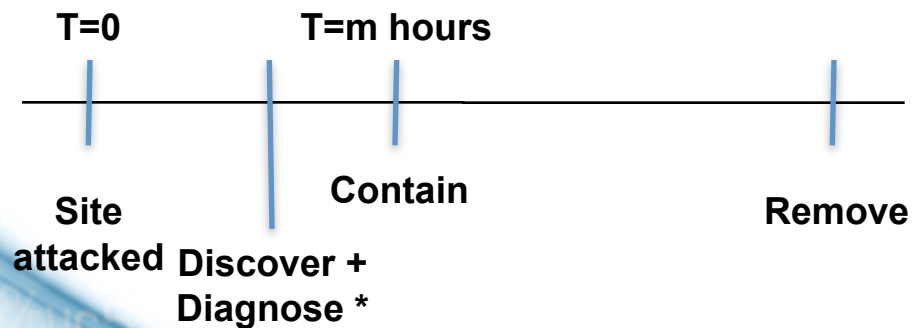
Mod_antimalware Architecture



Without Mod_Antimalware



With Mod_Antimalware



* - No time lag between Discover, Diagnose and Contain with Auto-Containment enabled

Significantly reduce
reaction time
(m hours << n days)

Mod_antimalware Architecture

Apache module (IIS also).

Output filter: main function is `quarantine_filter()`.

Two versions: standard & lite (open-source)

Handlers:

- `statusz`: echo last config directives

- `configz`: modify config directives

- `topz`: echo most frequently accessed URLs

Mod_antimalware Configuration Directives

DasientSharedAuthKey your_key_goes_here

SetBlacklistRedirectMessage “This server is experiencing technical difficulties. Please come back later.”

BlacklistRedirectUrlPrefix /foo1.html
(e.g. quarantining directive)

RemoteDirectivesFile /etc/apache2/
antimalware_remote_directives.conf



Mod_antimalware Architecture

Quarantining Directives:

Blocking (mod_antimalware_lite)

vs.

Filtering (mod_antimalware)

Mod_antimalware Architecture: /statusz

Status

processID: 30051, parentProcessID: 21911

My Config time: Sun Jul 18 06:53:11 2010

Shared Config time: Sun Jul 18 06:53:11 2010

Shared Config Data: initial_data

QuarantineBytes 0 0

Total Bytes Parsed: 0

Quarantine Enabled: 0

BlacklistRedirectMessage set: 1

BlacklistRedirectMessage: This server is experiencing technical difficulties. Please come back later.

BlacklistRedirectUrlPrefix: /foo1.html

Mod_antimalware Architecture: /configz

processID: 9600, parentProcessID: 21911

Shared Config Data Before:

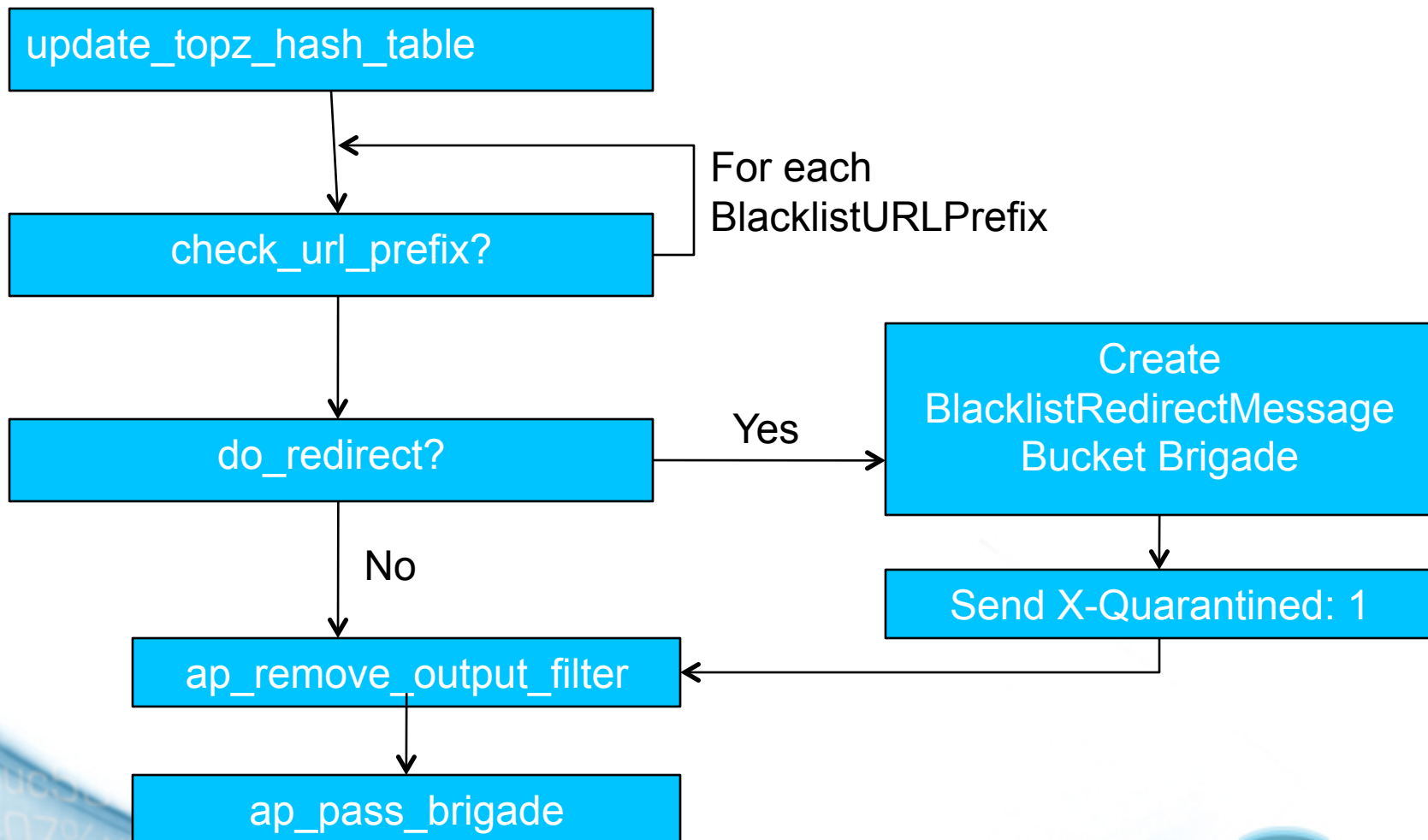
Shared Config Data After:

Successfully written to file: /etc/apache2/antimalware/

Mod_antimalware Architecture: /topz

processID: 12769, parentProcessID: 21911
/hackday08/randomtags.py 1
/blog/articles/week-of-unix-tools/main.html 1
/topz 1
/statusz 1
/favicon.ico 3
/ 2

Mod_antimalware_lite Architecture: quarantine_filter()



Mod_antimalware Implementation

Authentication

Partner Center: mod_antimalware

You are logged in as Neil Daswani (neil+goog@dasient.com).

[Submit Domains](#) | [Blacklist Report](#) | [Monetization Tools](#) | [Sales](#) | [mod_antimalware_lite](#) | [API](#) | [Logout](#)

[Installation](#) | [Activation Keys](#) | [Quarantining Status](#)

Activation Keys:

| Webserver | DasientSharedAuthKey | Enabled? | Status | Enable/Disable | Version |
|-------------|---|----------|--------|--------------------------|------------------------------|
| gnupods.com | TnyMeRWJhtwuxyYaphFrX6S1Alt
L8L99qSSya6evUoUFvWurVvgeS | Y | OK | <button>Disable</button> | Apache Standard |

Quarantining Verification

Restart-free Reconfiguration (via shared memory) + Persistence



Future Work

(open-source projects available)

Virtual Host Support

Certificate-based mutual authentication

Automatic deployment of quarantining directives

Where to learn more

- mod_antimalware SourceForge Page:
<http://sourceforge.net/projects/modantimalware/>
- Dasient Home Page / Blog / Twitter:
<http://www.dasient.com>
<http://blog.dasient.com>
<http://twitter.com/dasient>

Where to learn more

- Neil's Home Page:
<http://www.neildaswani.com>
- Stanford Security Certification Program:
<http://bit.ly/90zR1y>

Where to learn more



Foundations of Security:
What Every Programmer To Know
by Neil Daswani, Christoph Kern, and
Anita Kesavan (ISBN 1590597842)

Book web site:
<http://www.learnsecurity.com/ntk>

Free slides at:
<http://code.google.com/edu/security>

