Systems Thinking + Web Security

Michael Stone <mistine@akamai.com>
Q: Why InfoSec?  
A: Storytime...

2007-2009:

one laptop per child

2009-present:

search: "nortel espionage", "stuxnet", "aurora"
Q: Can system safety improve web security?

Q: ...or vice versa?

I claim:
I claim:
Motivation 1: Diagrams sure look similar...

EaSW, Ch. 8

Akamai:
Motivation 2: Definitions seem to align...

safety:
absence of accidents

accident:
unplanned + unacceptable loss event

"accident":
an unplanned + unacceptable loss event...
   ...potentially triggered by malicious activity.

∴ "Accidents" ⊆ Accidents
∴ Safety ⇒ no "accidents"
Problems:

1. No credible documentation of legacy safety constraints.

2. Control is non-hierarchical.

3. The system changes fast.

Solution:

rubric + examples (+ research).

Motivation:

People need digestable training materials.
At the coffee shop where Alice is browsing Bob's e-commerce site...

we are relying on Alice and Bob and every SSL CA...

to keep Alice's credit card number secret...

despite Mallory's snooping...

by correctly using HTTPS and X509.
Key Observation

To get security by way of safety, incorporate your **Adversary** into the **environment** and build a control system to suit.

Typical **goals**: secrecy, authentication, availability, access control, ...

Typical **adversary powers**: reading, writing, spamming, spoofing, parsing, unparsing, ...
Example

Login(Bob, K_{bob}) → Bob's EMail Server

Goals

- availability
- authentication
- secrecy
- access control
Q: Is this "safe"?

A: Depends on the Adversary's powers!

<table>
<thead>
<tr>
<th>Power</th>
<th>Affected Goal(s)</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>provide</td>
<td>secrecy</td>
<td>impersonate Bob</td>
</tr>
<tr>
<td>withhold</td>
<td>availability</td>
<td>DoS Bob</td>
</tr>
<tr>
<td>delay</td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>...</td>
<td>...</td>
</tr>
<tr>
<td>read</td>
<td>secrecy</td>
<td>steal passwd</td>
</tr>
<tr>
<td>write</td>
<td>authentication</td>
<td>guess passwd</td>
</tr>
<tr>
<td>spam</td>
<td>availability</td>
<td>lock Bob's acct</td>
</tr>
<tr>
<td></td>
<td>availability</td>
<td>DoS Server</td>
</tr>
</tbody>
</table>
Zooming out...

Principals are (smaller) systems: (i.e., "Alice" = Alice + desktop + browser + ...)

Protocols make untenable assumptions.

Conflicts of interest abound.

Finally, there are systemic risks...
Questions?

Michael Stone
<mistone@akamai.com>

http://mstone.info

(Thanks!)