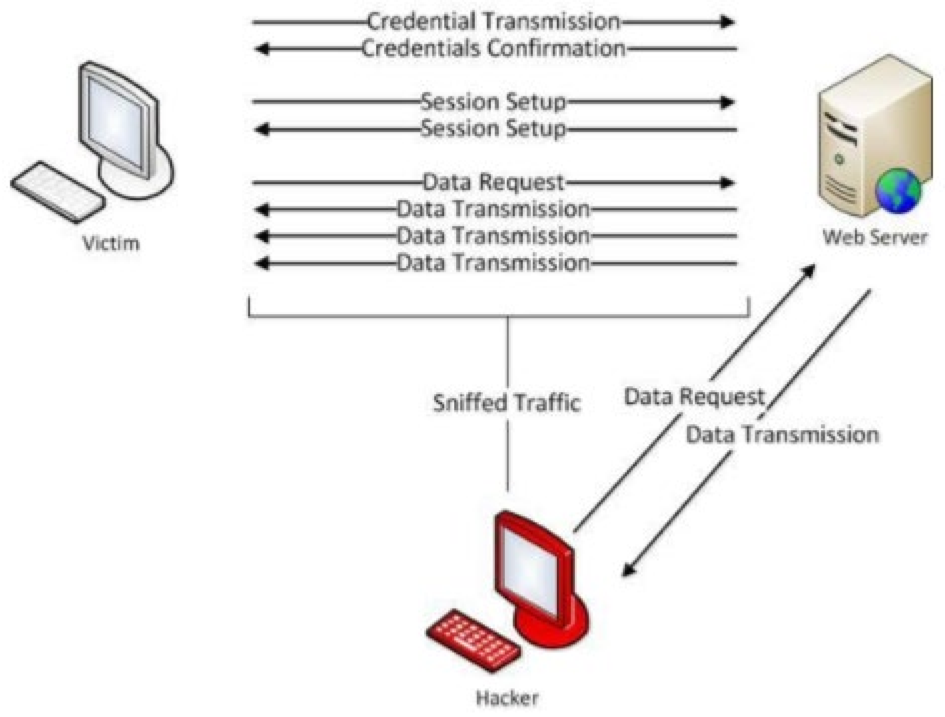
Session hijacking
Clickjacking
Waterhole

Session hijacking

- taking over an active TCP/IP communication session without the user's permission
- same access to resources as the compromised user
- Identity theft, Information theft, stealing sensitive



Types of session hijacking attacks

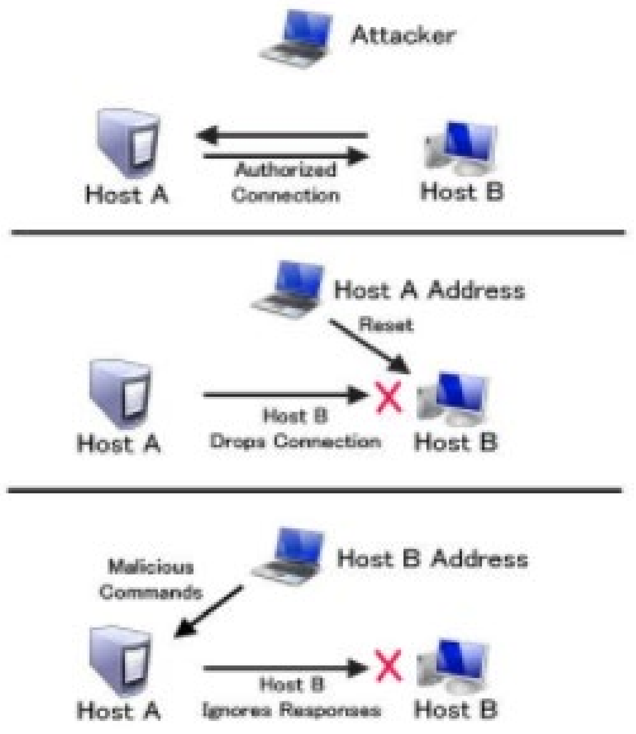
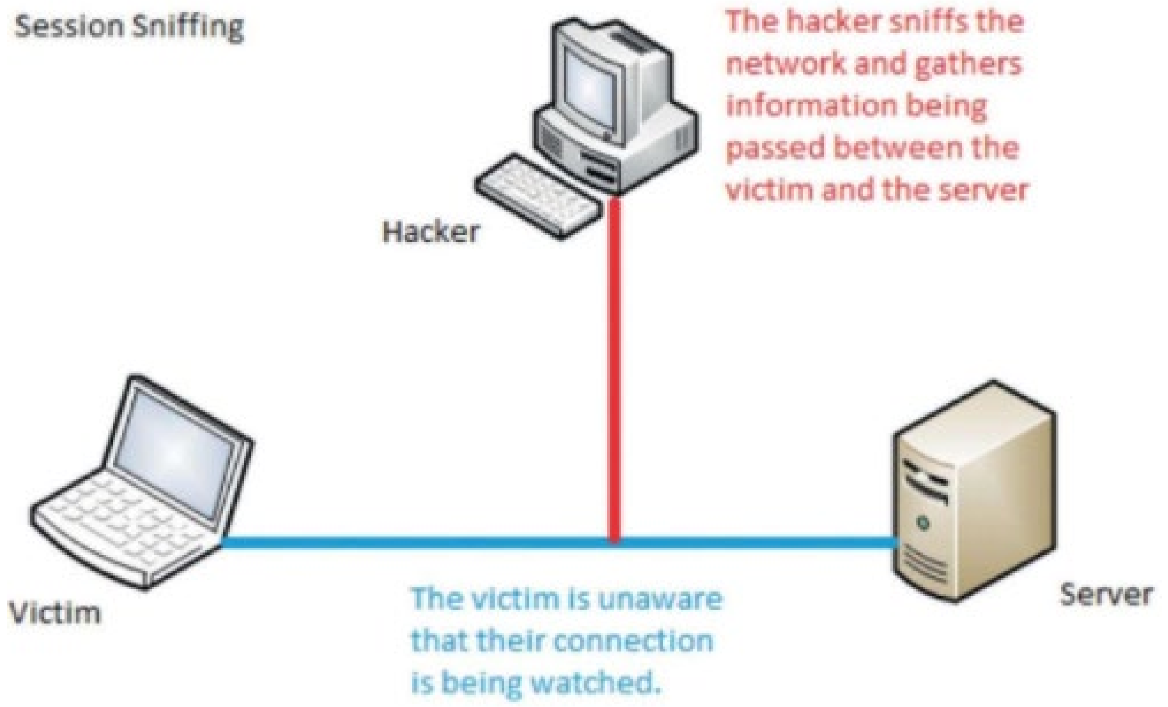
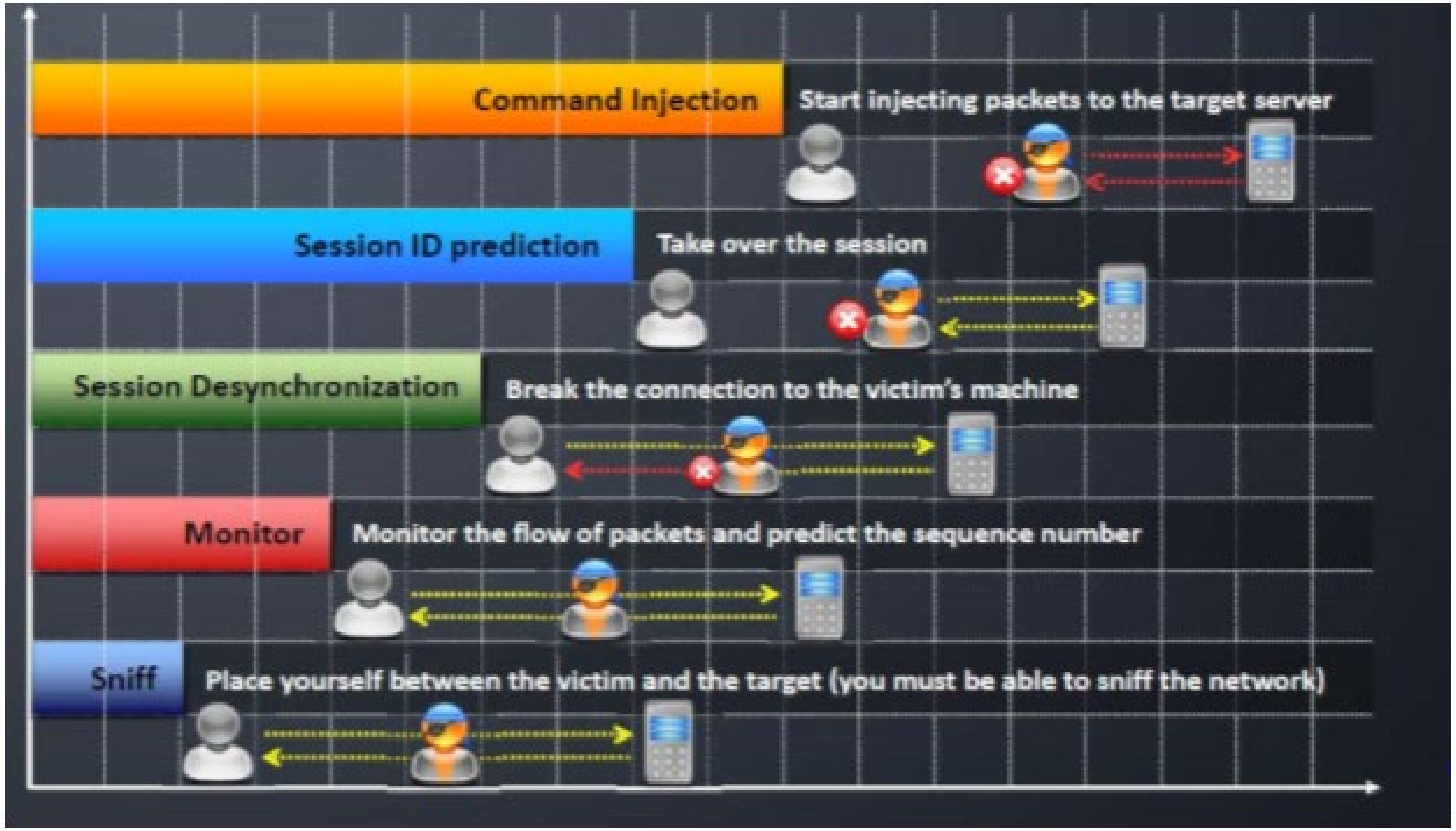


Figure 2: Session Hijacking


Session Sniffing
Hacker
The hacker sniffs the network and gathers information being passed between the victim and the server
Victim
Server
The victim is unaware that their connection is being watched.

Session Hijacking Process


Command Injection
Start injecting packets to the target server
Session ID prediction
Take over the session
Session Desynchronization
Break the connection to the victim's machine
Monitor
Monitor the flow of packets and predict the sequence number
Sniff
Place yourself between the victim and the target (you must be able to sniff the network)

Application Level Hijacking

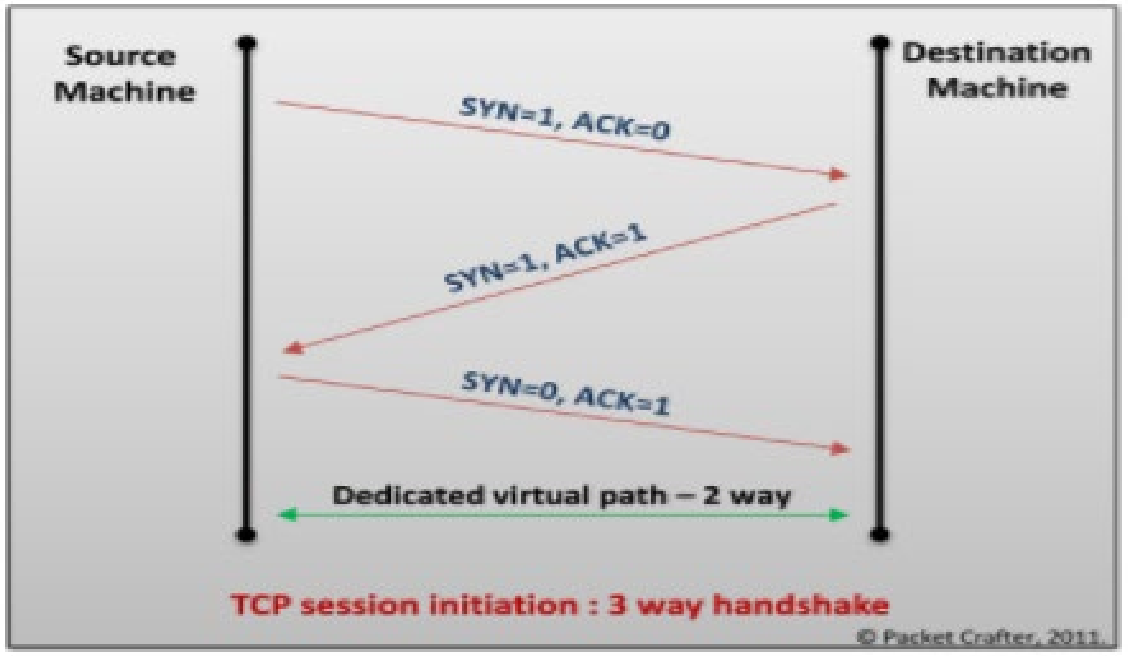
- Man in the middle attack:
- Cross-site scripting
- Using Proxy:
- Man-in the–Browser:
- Session Replay:

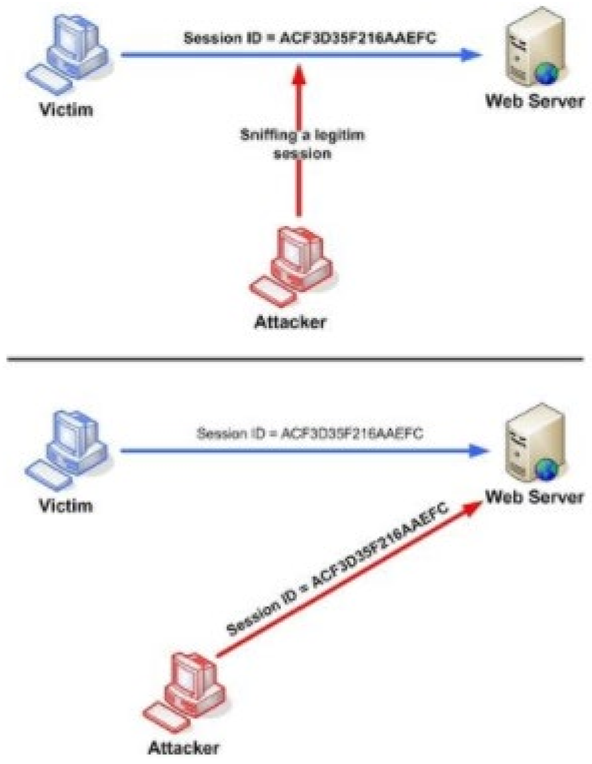
Network Level & Application Level

Activity

- Spoofing vs. Hijacking   pg 125
- TCP Concepts: Three-Way Handshake   pg 126

Network or TCP Session Hijacking



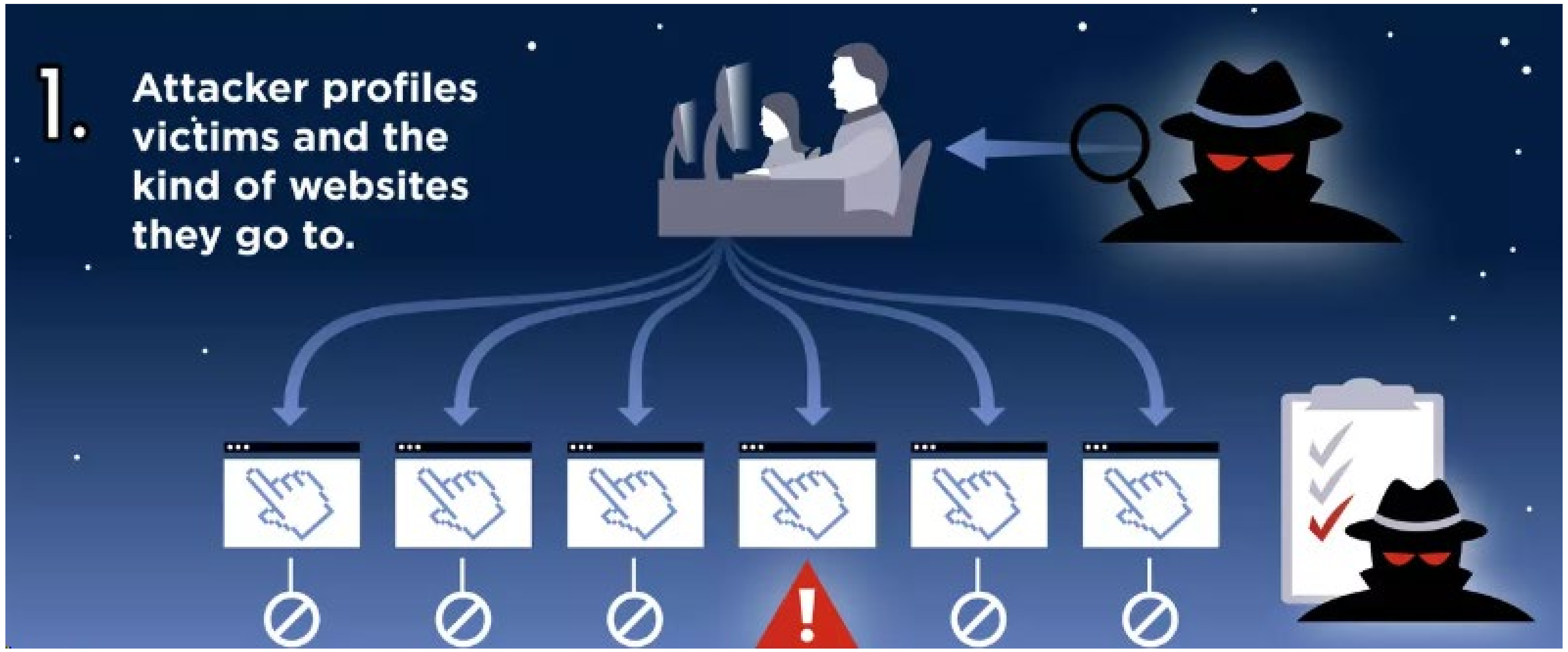

Victim
Session ID = ACF3D35F216AAEFC
Web Server
Sniffing a legitim session
Attacker
Victim
Session ID = ACF3D35F216AAEFC
Web Server
Session ID = ACF3D35F216AAEFC
Attacker

TCP session hijacker is to create a state where the client and server are unable to exchange data;
enabling him/her to forge acceptable packets for both ends, which mimic the real packets.
Thus attacker is able to gain control of the session.

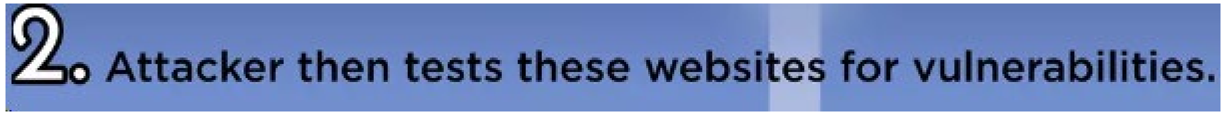
Counter Measures:

- Using secure protocols instead of clear text protocols like HTTP, FTP. Telnet, Rlogin, etc.
- Encrypting session id will increase the complexity of the session id prediction.
- Sending session id over SSL.
- Use long random numbers for session id.
- Implement timeout for the session when the session is logged out, or session id expires.
- Having different session id for each page.
- Use switches rather than hubs.
- Ensure server side and client side protection software.
- Use IDS for detecting ARP spoofing/Poisoning.
- Do not click on suspicious links.
- Check the web application for all errors.
- Using IPSec is a valid defence mechanism.

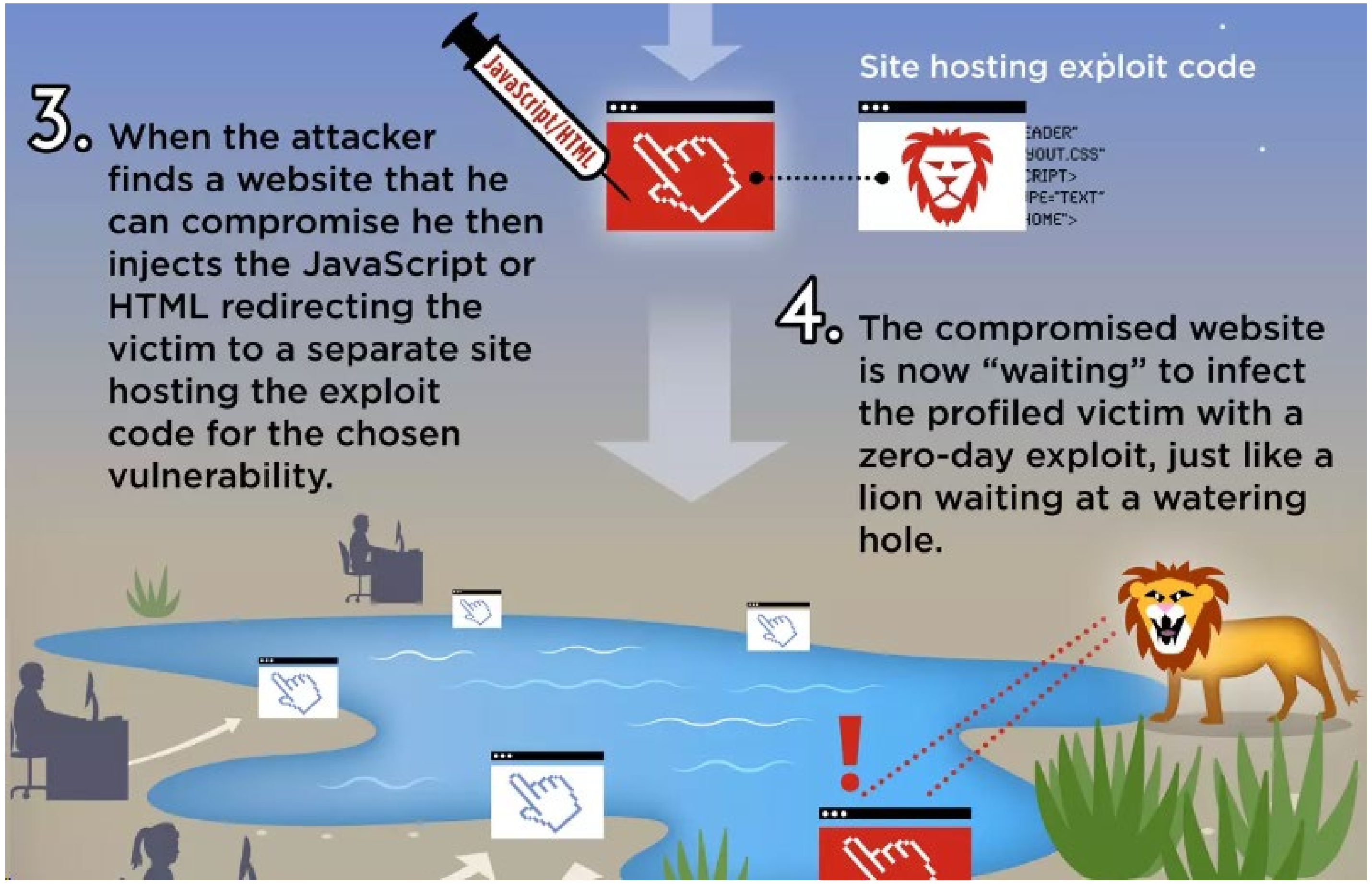
Waterhole Attack


1.
Attacker profiles victims and the kind of websites they go to.

Waterhole Attack



2. Attacker then tests these websites for vulnerabilities.

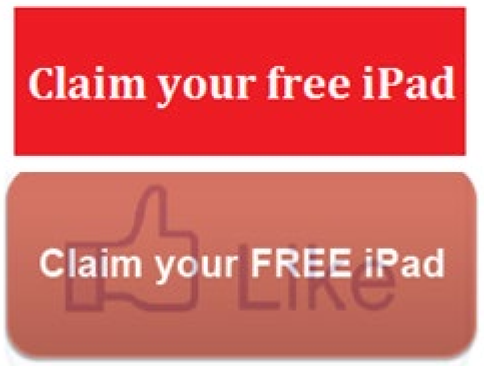


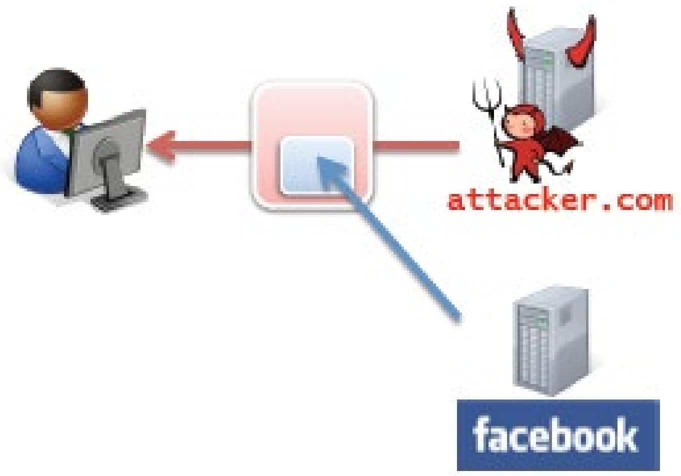
3. When the attacker finds a website that he can compromise he then injects the JavaScript or HTML redirecting the victim to a separate site hosting the exploit code for the chosen vulnerability.

4. The compromised website is now "waiting" to infect the profiled victim with a zero-day exploit, just like a lion waiting at a watering hole.

Likejacking

- The user can be tricked into clicking 👍 Like button, on an attacker's website
- User visits attacker.com
- Like button hidden behind another button





Clickjacking: Definition

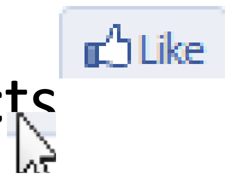
- **Prerequisite**: Multiple mutually distrusting applications sharing the same display, and having permission to manipulate each other’s visual appearance
- Attacker comprimises context integrity of another app’s UI components
  - Temporal Integrity
  - Visual Integrity

Types of Context Integrity

Visual Integrity

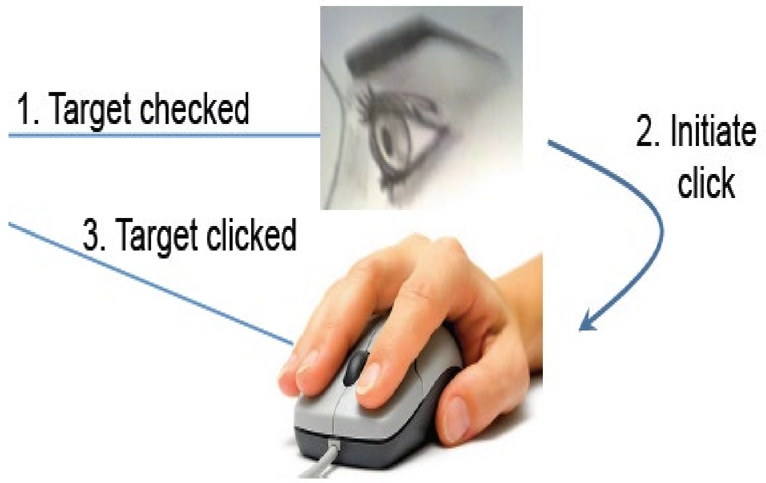
- What the user sees, is actually what is present
- No transparent, overlayed objects
- Eg should be visible

should be visible



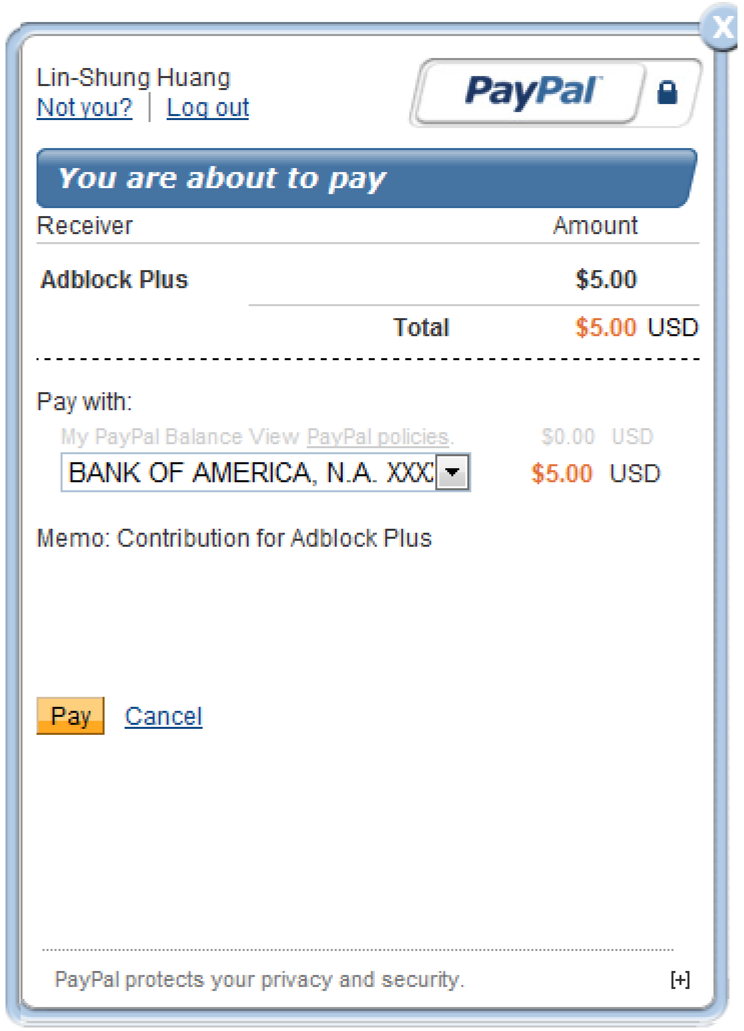
Temporal Integrity

- State of the UI between *time of user checking* and *the time of initiating the click*, remains the same



Compromising Visual Integrity

- Hide the target
- **Partial Overlays**



Lin-Shung Huang
Not you? | Log out

You are about to pay

Receiver		Amount
Adblock Plus		$5.00
	Total	$5.00 USD

Pay with:

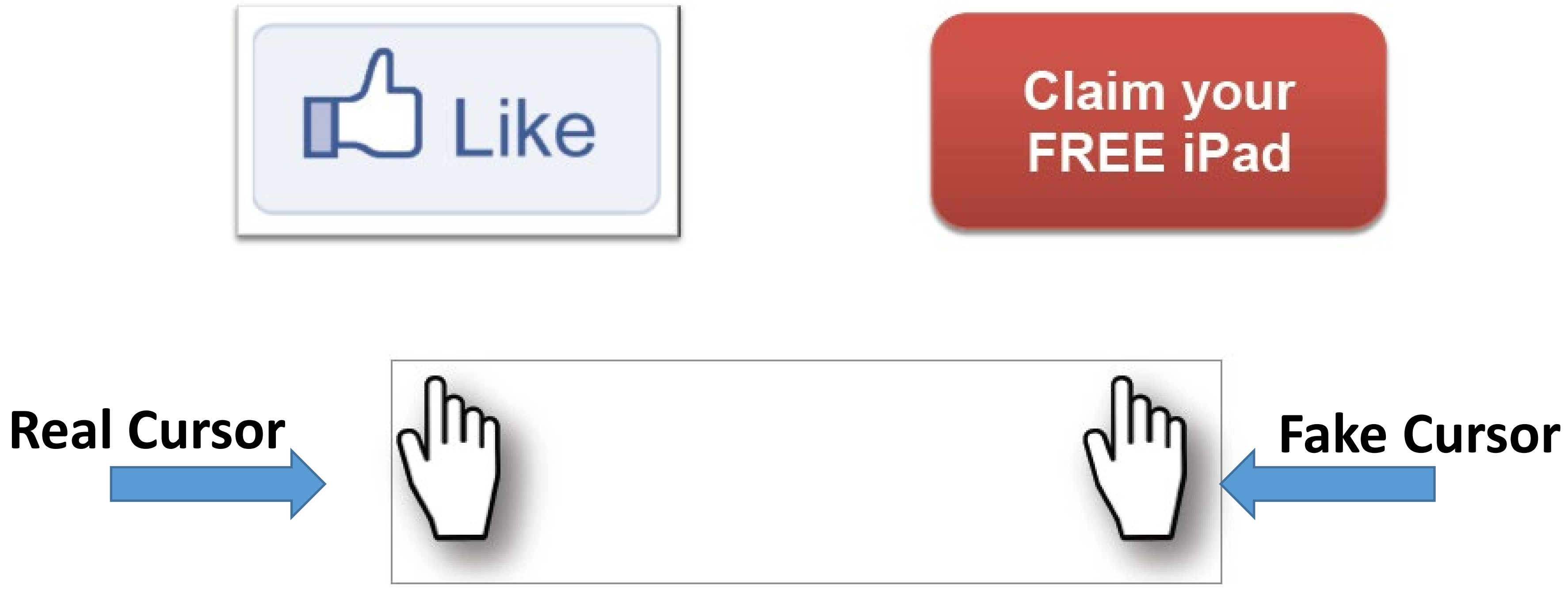
My PayPal Balance View PayPal policies.	$0.00 USD
BANK OF AMERICA, N.A. XXX	$5.00 USD

Memo: Contribution for Adblock Plus

PayPal protects your privacy and security.

Compromising Visual Integrity

- Multiple cursor feedback known as *cursorjacking*



Compromising Temporal Integrity

- Bait and switch: As mouse comes near “Claim you..” button, Like moves to take it’s location before the user realizes it

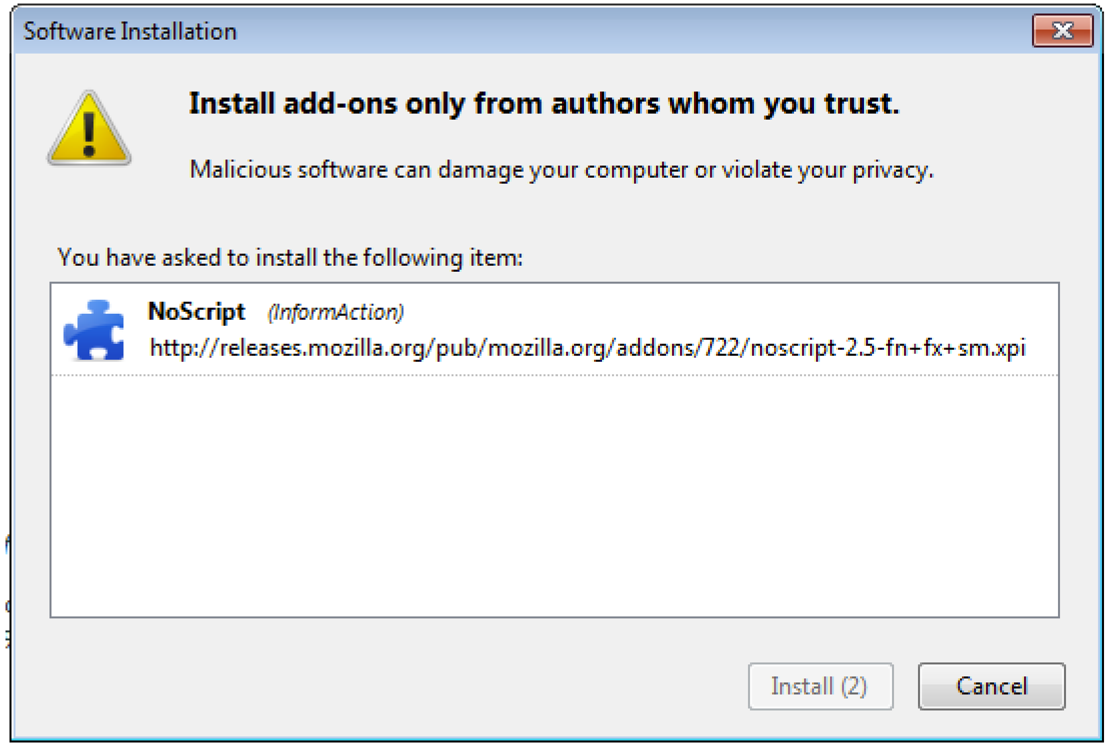
Claim your FREE iPad
Like

Existing Defences

- User confirmation
  - Degrades user experience
- UI randomization
  - Unreliable & not user-friendly. (Multi-click attacks)
- Framebusting (X-Frame-Options)
  - Incompatible with embedding 3rd-party widgets
- Opaque overlay policy
  - Breaks legitimate sites
- Visibility detection on click
  - Allow clicks only on elements that are visible

Protecting temporal integrity

- Imposing a delay after displaying a UI
  - Annoying to users



Software Installation

Install add-ons only from authors whom you trust.

Malicious software can damage your computer or violate your privacy.

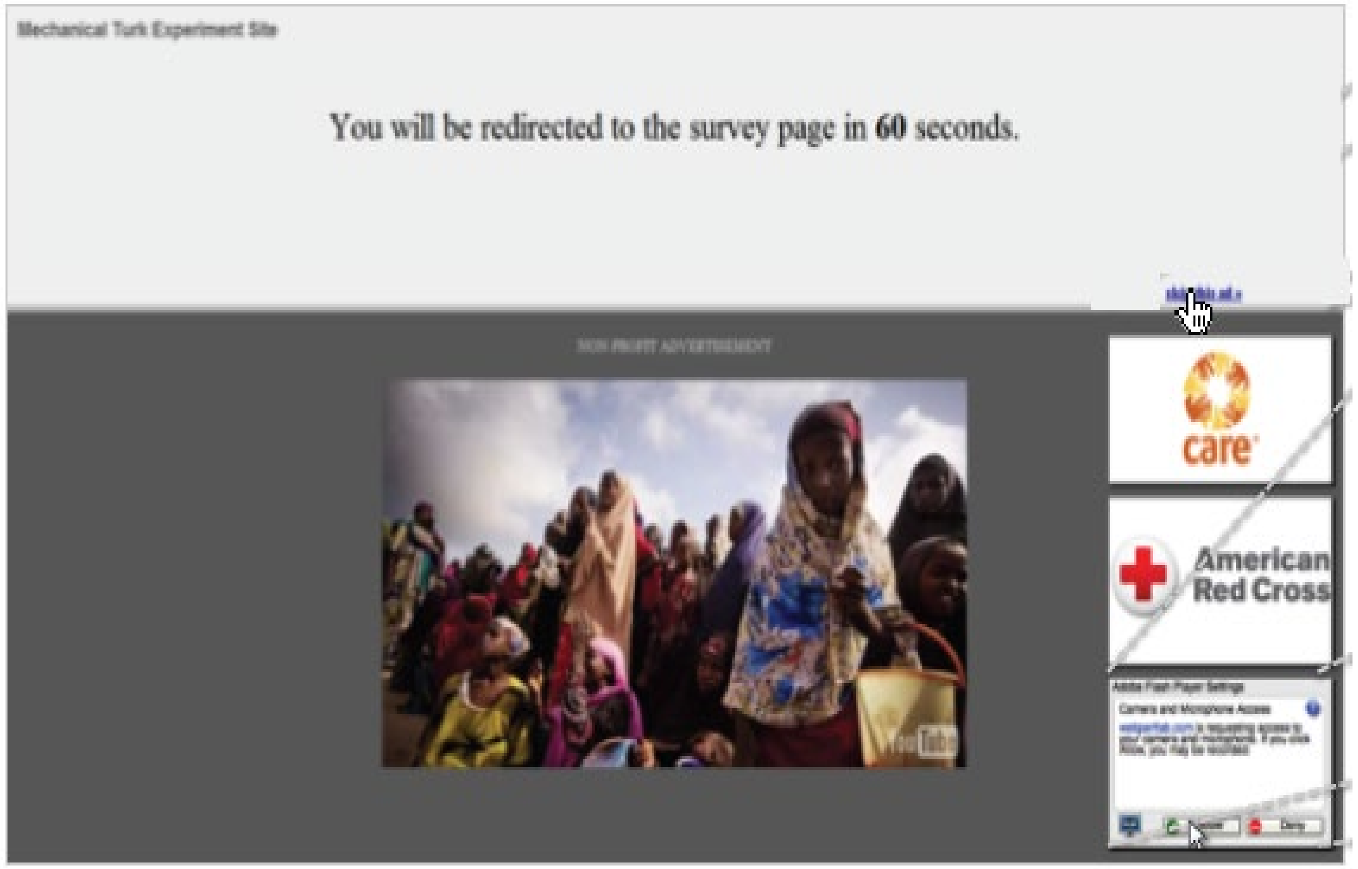
You have asked to install the following item:

NoScript *(InformAction)*
http://releases.mozilla.org/pub/mozilla.org/addons/722/noscript-2.5-fn+fx+sm.xpi

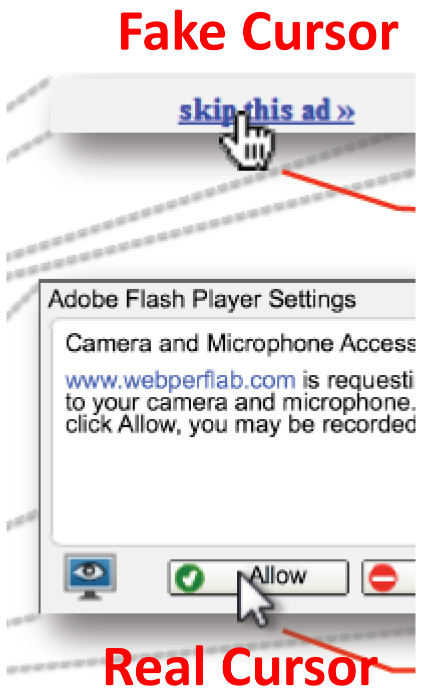
New Attacks Demonstrated

- Authors conducted new exploits using Clickjacking & with and without their own patches using Amazon Mechanical Turks
- Reported the effectiveness of the attack
- Attacks:
  - Accessing user's webcam: **Attack success: 43%**
  - Stealing user's email: **Attack success: 47%**
  - Revealing user's identity: **Attack success: 98%**

Accessing user’s webcam


Mechanical Turk Experiment Site

You will be redirected to the survey page in **60** seconds.

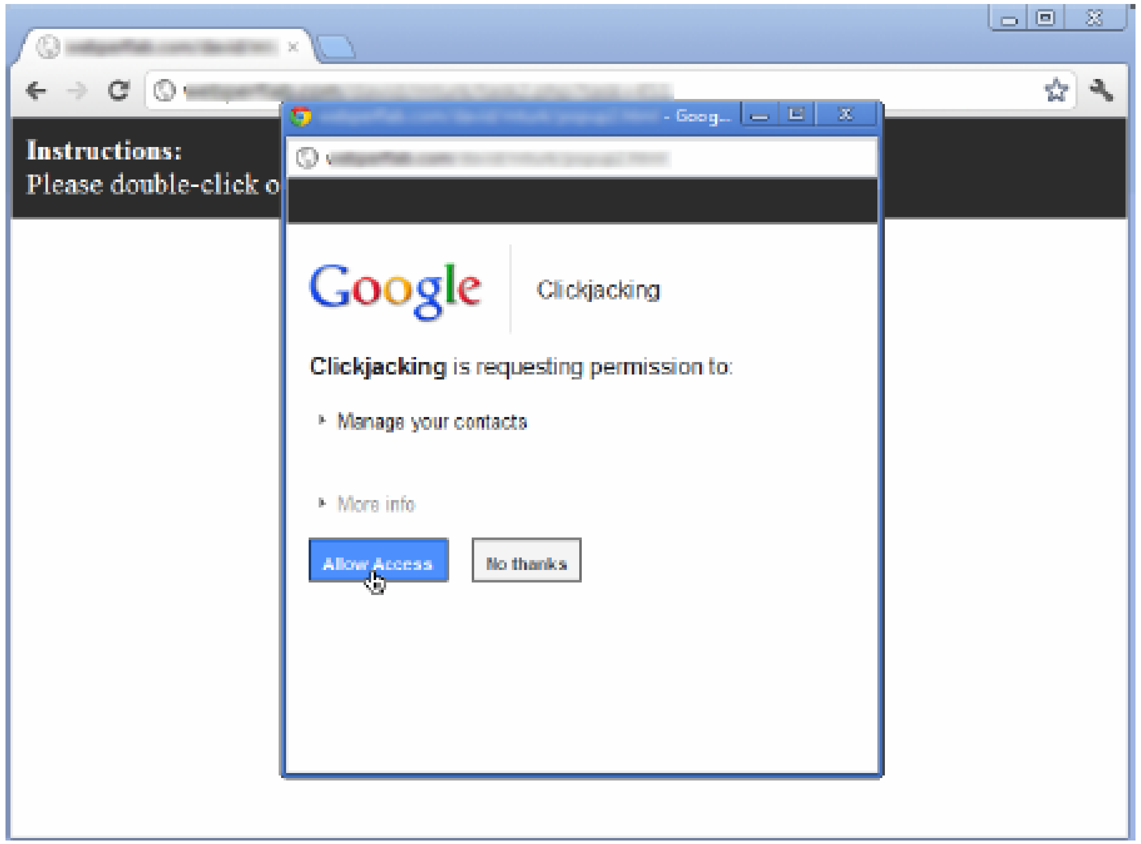


Adobe Flash Player Settings

Camera and Microphone Access

www.webperflab.com is requesti
to your camera and microphone.
click Allow, you may be recorded

Stealing user’s email



Instructions:
Please double-click o

Clickjacking

Clickjacking is requesting permission to:

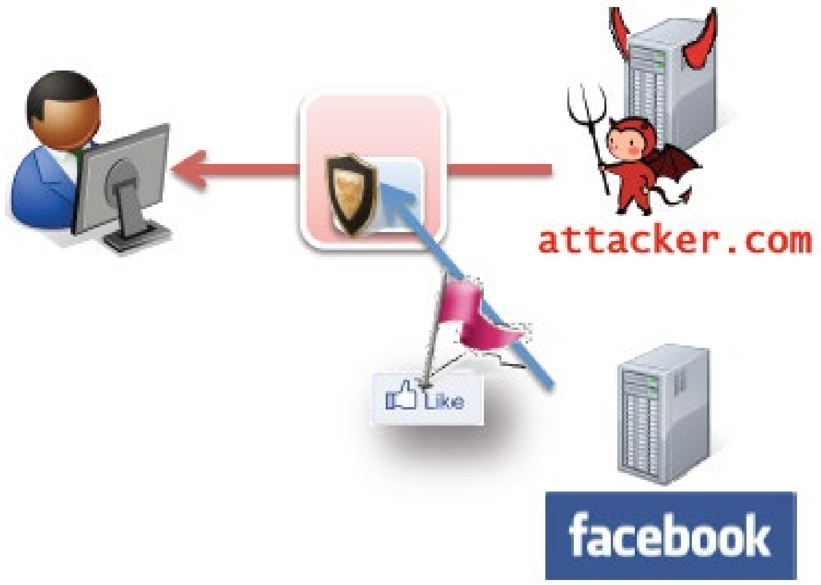
- Manage your contacts
- More info

InContext Defence

- Design Goals:
  - Should support 3rd party object embedding
  - Should not have to prompt users for actions
  - Should not break existing sites
  - Should be *resilient* to new attacks

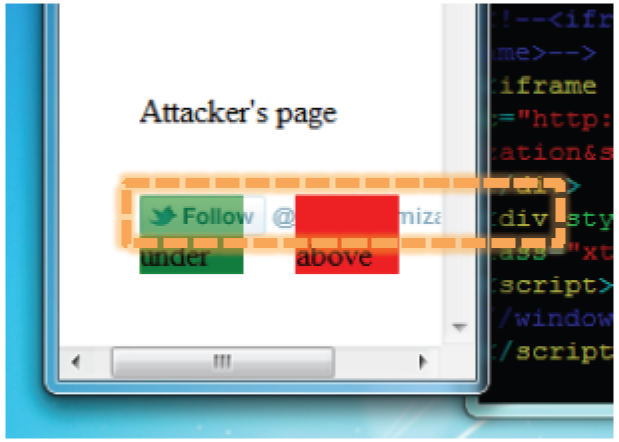
Basic Idea

- Techniques to ensure user is always *InContext* of the sensitive UI in interaction
- Websites can indicate their sensitive UI
- Browsers can *enforce* context integrity rules on these sensitive UIs



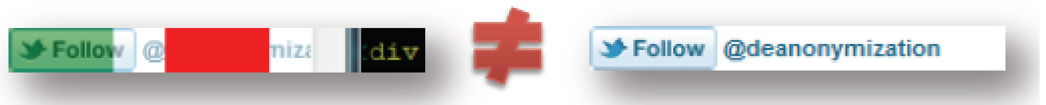
Ensuring visual integrity of target

- OS can compare the screenshot of sensitive UI with the reference bitmap provided
  - 30ms overhead on click processing



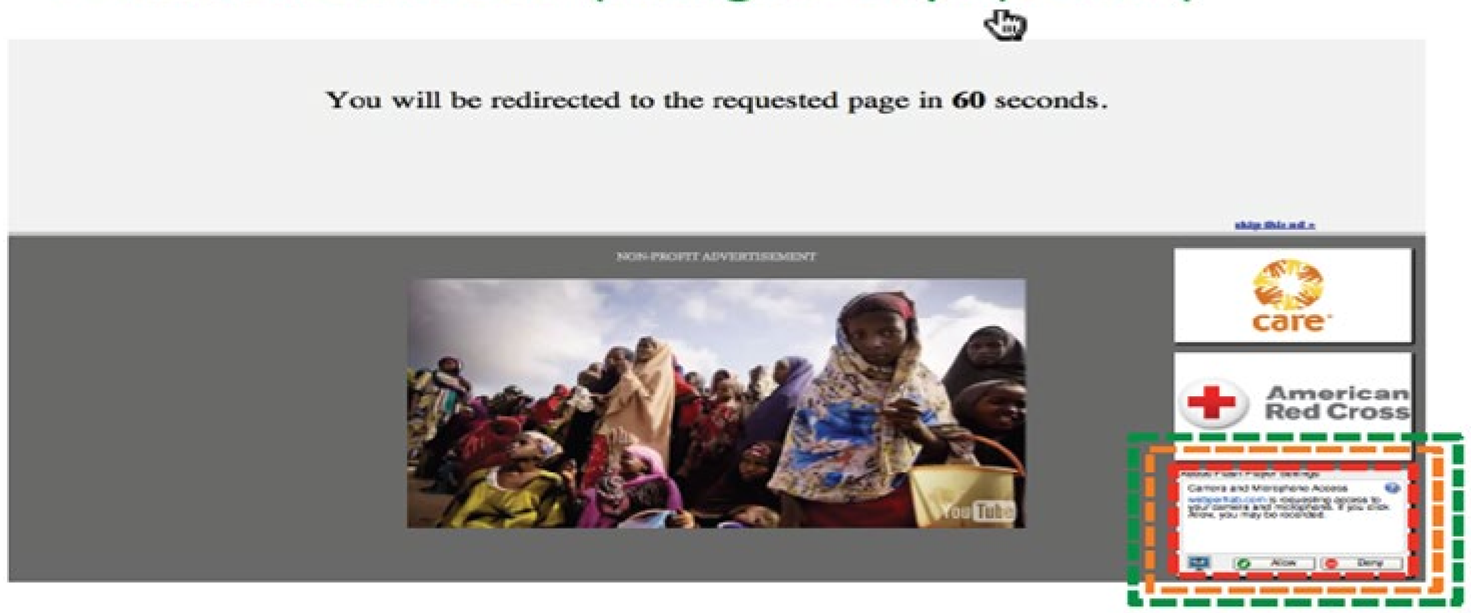
What is displayed (OS screenshot)

What should be seen (Reference bitmap)



Ensuring visual integrity of pointer

- Remove cursor customization
- Freeze screen
  - Attack success: 43% -> 15%
  - Attack success (margin=10px): 12%
  - Attack success (margin=20px): 4%

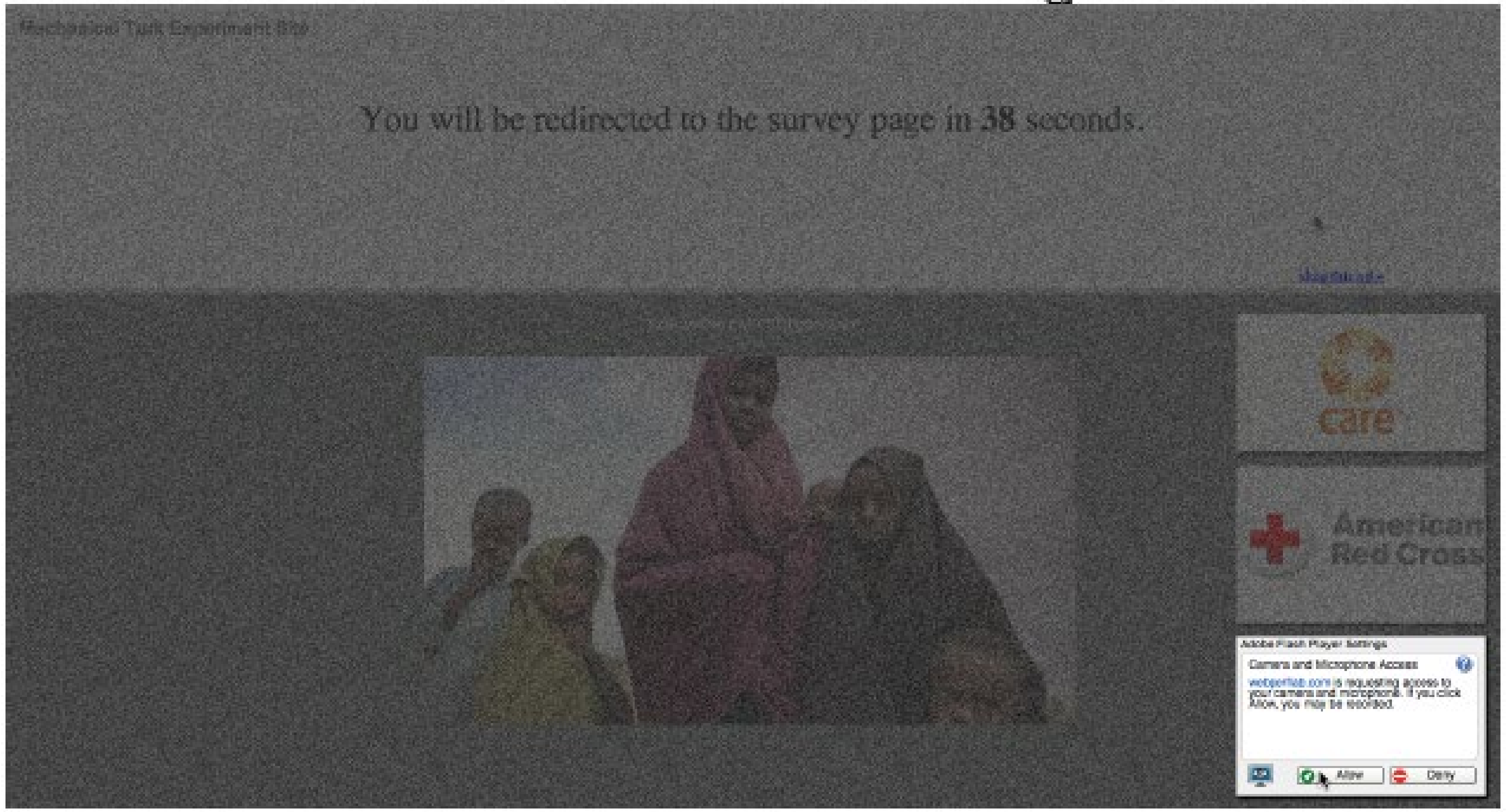


You will be redirected to the requested page in **60** seconds.

skip this ad »
NON-PROFIT ADVERTISEMENT
care
American Red Cross
Adobe Flash Player Settings
Camera and Microphone Access
webpefab.com is requesting access to your camera and microphone. If you click Allow, you may be recorded.
Allow
Deny

Ensuring visual integrity of pointer

- Lightbox effect around target on pointer entry



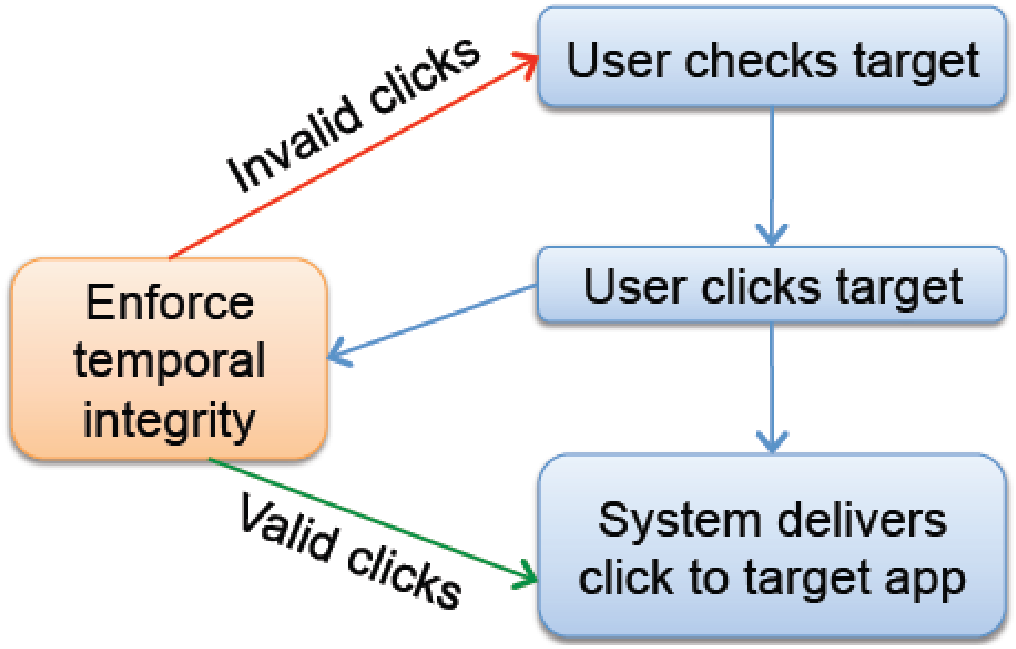
Mechanical Turk Experiment Site

You will be redirected to the survey page in **38** seconds.

webperflab.com is requesting access to your camera and microphone. If you click Allow, you may be recorded.

Ensuring temporal integrity

- **UI Delay**
  - On a visual change, all buttons are inactive for a certain time
- **Pointer Re-entry:**
  - On a visual change, invalidate clicks till pointer re-enters the UI


User checks target
User clicks target
System delivers click to target app
Enforce temporal integrity
Invalid clicks
Valid clicks