Web Security

Dawn Song
dawnsong@cs.berkeley.edu

Mid-term Questionnaire Summary (I)

• Optional readings
  – You don’t have to read them
• Paper summaries
  – Should not take too much time
  – No homeworks, so load is balanced
  – Bullet form is ok
  – Due before class?
  – Summaries on-line?
  – Feedback on summaries?
  – Readings will be reduced in 2nd half of semester
    » Give time for project
• Speed
  – People have diverse background, so it’s difficult to satisfy everyone at the same time
  – Thanks for understanding

Mid-term Questionnaire Summary (II)

• Guest lecture
  – Many students really like the idea
  – We’ll have a few more guest lectures
  – Would have liked more discussions:
    » Prepare your questions
• Discussions
  – Many find exciting & insightful
  – More people need to participate!
    » No pressure
    » Don’t be shy :)!
    » Try to contribute with your thoughts/questions
    » Try to bring your comments to OpenMic
• Students select topics
  – Let me know & we’ll try to accommodate if there’s time
Project Proposal

• Mostly fine with topics
  – Scott & Craig: come see me after class

• Many lack timeline
  – Include timeline & resubmit by Oct 22

• Milestone: due Nov 14

• Poster session: Dec 6, 2:30-4:30pm
  – In conjunction with CS261

Browser-OS Analogy

• OS
  – Resource management
  – Layer of abstraction
  – Isolation

• Browser-platform
  – What resources does browser-platform manage?
    » OS analogous?
  – What abstractions does browser-platform provide?
    » OS analogous?
  – What properties should browser-platform ensure?
    » OS analogous?

Straw-man Approaches

• VMWare Web browser appliance
  – A check-pointed image of Firefox browser on Linux
  – Disadvantages?
• What about running each URL in a separate VM?
Tahoma Architecture

- Trust model & principles
  - Web applications should not be trusted
    - Web application = Browser instance + web services
    - Isolation: each browser instance in VM
  - Web browsers should not be trusted
    - Isolate browsers from rest of the system
    - Network policy & reverse firewall
  - Increase visibility & control over downloaded web applications
    - Web applications should be visible to users like desktop applications

Manifests

- Tahoma web applications are first-class objects
  - Explicitly defined & managed
- Manifests
  - Digital signatures authenticating web service
  - Browser policy: code to run in browser instance
  - Network policy: internet access policy to be enforced by reverse firewall
- A paradigm for mobile code
  - Signature + code + sandbox policy
**Browser Operation System (BOS)**

- TCB for Tahoma browsing system
- Multiplexes virtual screens of each browser instance into physical display
  - Trusted border
- Enforce network policies for each instance
- Store state for associated browser instance
  - Bookmarks, manifests
- Inter-application communication
  - Fork, BinStore, BinFetch

---

**Tahoma Implementation**

- Xen VMM in Linux
- BOS, BOS Kernel & tiny proxy implemented as domain0 VM
- Browser instance run on Xen VM
- Window manager aggregates virtual screens on physical screen
- Browser modifications
  - Linking to libQT to access Tahoma graphics subsystems
  - Using browser-call to access remote services
  - Using browser-call for new functions, e.g., fork

---

**Discussions**

- Advantages of Tahoma
  - What common attacks does Tahoma prevent?
- Disadvantages of Tahoma?
  - What kinds of attacks does Tahoma fail to prevent?
- How does Tahoma compare with SFI/XFI?
- Does Tahoma provide a trusted-path btw user & web service? Why?
Open Mic

• Anything else you thought that’s really clever in the papers?
• Anything else you didn’t like about the papers?
• Any other unclear points about the papers?
• Other comments/remarks to share?

Summary

• BrowserOS

• Next class:
  – Mashup OS
  – XSS