Simplified Web Transaction

Client Browser

Request

Web Server

Response

Client Side Application, Running with Server Provided Code & Data

Server Side Application, Running against User Provided Data
Browser Anatomy

- HTTP
- HTML Parsing (or other ML)
- Plug-ins
- ActiveX
- Java
- Scripting (e.g., JavaScript, VBScript)
- Rendition
- Helper

- Render
- Interpret
- Parse
- Transport
Execution Environment

Response

Browser

Render

Built-in

Plug-in

ActiveX Control

Code

Helper Application

Code

Resources
Risks of Mobile Code

- Execution environments are designed to offer mobile code access to local resources: windows, files, network sockets, etc.
- Environments are often set up to run mobile code automatically, malicious code may:
  - gain unauthorized access to resources
  - manipulate resources covertly
  - ignore local security policies/procedures
  - perform malicious actions using the user's identity
- Different mobile code technologies have:
  - widely varying degrees of access to resources
  - different kinds of security controls
Classes of Technical Controls

- **Cage** - constrain the code’s behavior (e.g., privilege or function) during execution
- **Filter** - examine code at an entry point and block or disable if deemed harmful
- **Signature** - execute code only if it is digitally signed by some trusted authority
- **Proof** - before executing the code, verify that the proof of its properties, conveyed with it, satisfies policy
- **Hybrid** - some combination of the controls above

*Mobile Code Security, Rubin & Geer, Internet Computing Vol. 2, No. 6*
Assessing Risk

Assessing the risk imposed by a particular mobile code technology begins with examining the code's context:

- access to resources: display, files, network, etc.
- security controls: how and when controls are imposed, how effective they are, how well they can be configured
- other countermeasures: effectiveness of external technical or policy countermeasures

Choosing whether to support and accept a particular mobile code technology must involve balancing its risks against the benefits it can provide.
Example Risk Categories

- High - Conveyed mobile code has broad functionality with unmediated access to computational resources
- Medium - Conveyed mobile code has broad functionality with controlled access to computational resources
- Low - Conveyed mobile code has limited functionality with controlled access to computational resources
- Plus accounting for the presence or absence of code signing

*DoD Mobile Code Policy Guidance, Money, Pentagon Memo, Nov. 7, 2000*
Technology Related Risks

- Portable Document Format (PDF)
- JavaScript and VBScript
- New Media Plug-ins
- Java
- PostScript
- ActiveX

Low
Medium
High
Guidance Overview

- NIST guides have been concerned with accessing public as opposed to private or sensitive information.
- Focus has been on protecting government systems from mobile code threats:
  - Deploy the lowest risk mobile code technology on Websites
  - Disable risky mobile code technologies on browser
  - Apply technical and other controls to mitigate risks
Summary

“Touching the browser” is a bit of a misnomer – we regularly affect the client side by serving acrobat, word, and other content.

Different technologies affect the browser differently and new technologies are continually on the horizon, making it difficult to pick a winner.

Ultimately, agencies are left with the decision as to how best to interface technologically with citizens.
Further Information

Computer Security Resource Center Guides:

Guidelines on Active Content & Mobile Code

Guidelines on Securing Public Web Servers

Security for Telecommuting & Broadband Communications

Email: Jansen@NIST.Gov