CSI NetSec 06
THE PRACTICE OF INFORMATION SECURITY

Bank Hacking Live!

Ofer Maor

CTO, Hacktics Ltd.

ATC-4, 12 Jun 2006, 4:30PM

HACKTICS

CSI COMPUTER SECURITY INSTITUTE Conferences

Agenda

- Introduction to Application Hacking
- Demonstration of Attack Tool
- Common Web Application Attacks
- Live Bank Hacking Demonstration
- Questions & Answers

Introduction to Application Hacking

Overview

- Today, most organizations create, use and externalize distributed applications implementing business processes.
- The increasing numbers of such applications combined with the improved security in the infrastructure layer drives hackers to turn to application attacks.
- According to Gartner, over 75% of attacks today take place in the application layer.

What Is Application Hacking?

- Taking advantage of application-level vulnerabilities to attack the site
- Attacks relate to the semantics and meaning of application messages, such as HTTP requests, SQL Queries or proprietary requests.
- Differs from infrastructure attacks focusing on identifying unauthorized services (port scanning) and abusing known vulnerabilities.

Application vs. Infrastructure

- Not easily replicated (no script kiddies!), though still easily exploitable
- Target the organization's core business operations rather than technology
- Allows launching direct attacks rather than needing to break several circles of defense
- Used by attackers with specific agenda (criminals, industrial espionage, etc.).

Application Vulnerabilities Mitigation

- No prepared patch to easily deploy
- Fixing the vulnerability requires recoding, turning it into a costly procedure
- Design Mistake Fix Cost Increase (Gartner):
  - 1x – During Design
  - 6.5x – During Development
  - 15x – During Testing
  - 100x – After Deployment

Technical vs. Logical

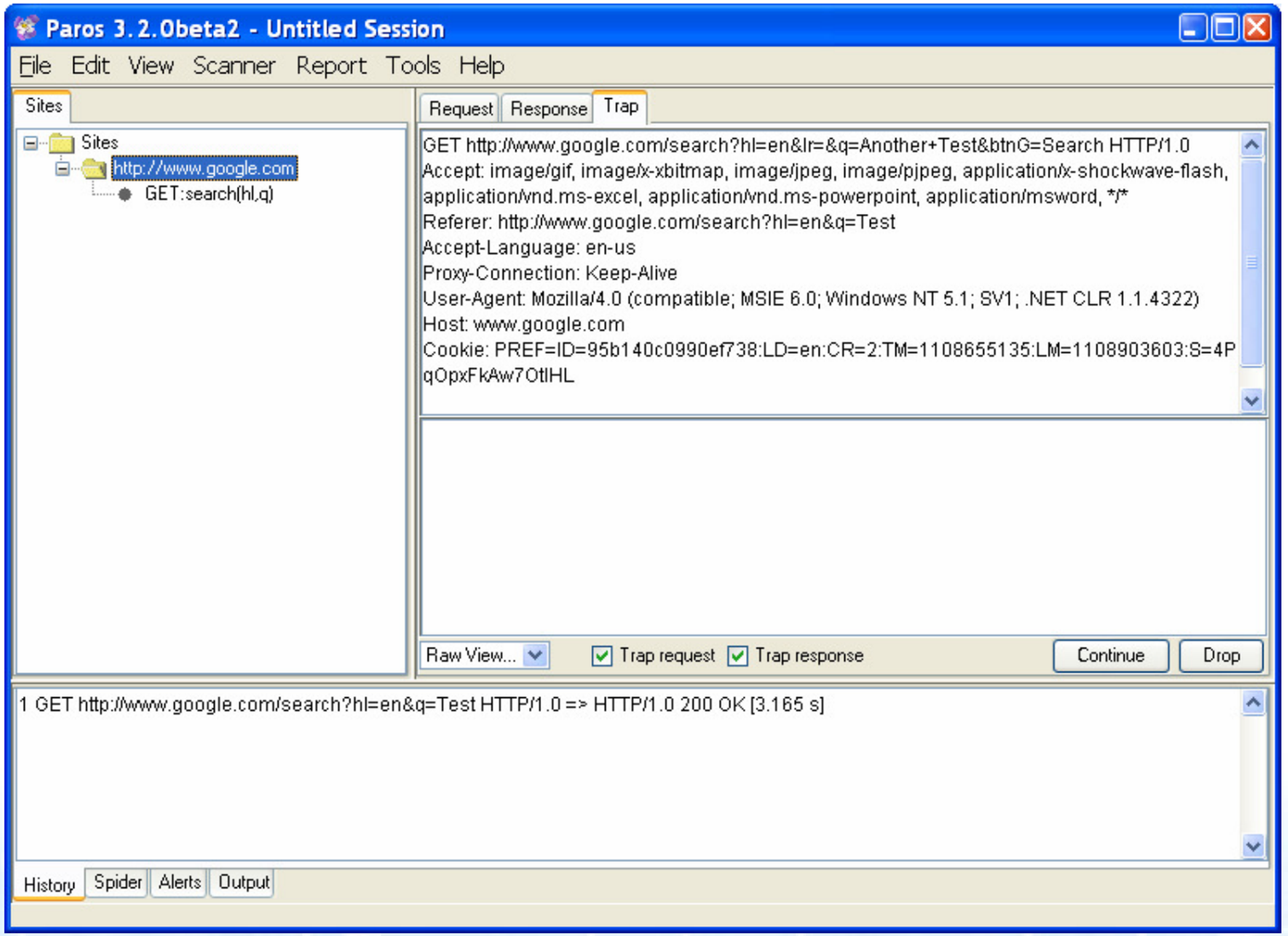
- Technical flaws relate to the specific technical implementation of the application
- Logical flaws relate to the way business processes were developed, unrelated to the development infrastructure
- New security features added to development infrastructure help decrease the number of technical flaws, whereas logical flaws are stil a prominent problem

Web Application Penetration Tool

Application Hacking Techniques

- Applications expect the *client* to behave in a certain predefined manner (only *user* controlled data is validated)
- The *client*, however, can be easily controlled by the malicious user (attacker)
- Easily done using friendly GUI based tools
  - Interactive Interception Proxies
  - Browser Plug-ins
  - Etc.

Interception Proxy Demo



Common Web Application Attacks (With Live Demo!)

Passive Reconnaissance

- Understanding the Application
- Requests Monitoring
- Structure & Flow Mapping
- Searching Code for Comments
- Identifying Development Infrastructure
- Retrieving Internet Resources
- Google Hacking

Active (Malicious) Reconnaissance

- Generate Exceptions & Errors
- Unreferenced URLs
  - Default Components
  - Administrative Interfaces
  - Configuration/Log Files
- Source Code Disclosure
  - Known Vulnerabilities
  - Backup/Old Files
  - File Access Components

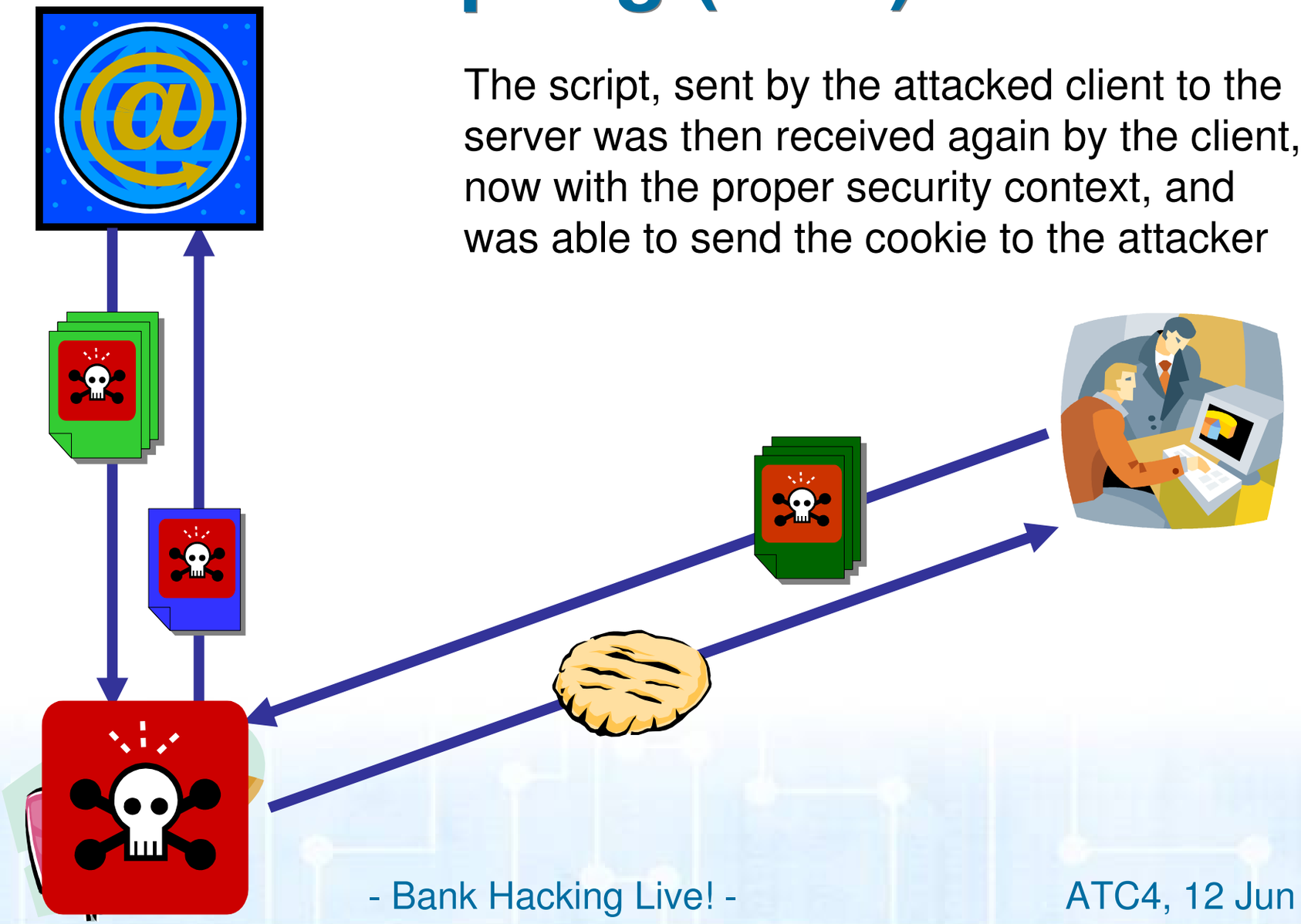
Parameter Tampering

- Basic, most simple form of application attack
- Directly targeting the business logic
- Does not require deep technical knowledge
- Takes advantage of developer assuming parameters retain their predefined values in
  - Links
  - Hidden Fields
  - Fixed Values
  - Etc.

Scripts Injection/Cross Site Scripting

- Most common web application vulnerability
- Used to bypass browser security in order to launch malicious scripts in the right context
- Performs an HTML injection of a JavaScript or VBScript on returning data
- Allows attacker to steal cookie information, steal data, execute operations on behalf of user, perform advanced phishing, etc.

Cross Site Scripting (XSS)



The script, sent by the attacked client to the server was then received again by the client, now with the proper security context, and was able to send the cookie to the attacker

Flow Bypassing (Forceful Browsing)

- Common Logical Attack
- Useful against step-based applications such as wizards or redirection-based applications
- Allows attackers to overcome specific authentication or authorization mechanisms

SQL Injection

- Most powerful web application attack – targeting the data itself
- Takes advantage of common usage of Dynamic SQL Queries
- Allows an attacker to maliciously modify the query sent by the application to the server
- Using this attack it is possible to bypass authentication, access sensitive data, modify data, cause DoS or takeover the server.

Thank You

Q & A

www.hacktics.com