

# Penetration Testing and the Metasploit Framework

Iowa State IASG

September 14, 2010

Jonathan Cran

# Introduction

- Jonathan Cran
  - Former CISSP
  - Security Consultant (Rapid7, LLC)
  - QA Engineer (Metasploit)
- Talk is split into two parts
  - Security Consulting / Penetration Testing
  - Metasploit Framework & Practical Ownage



**Eyeliner go like dis**

**right?**

**I good makeup artist**

# InfoSec Today

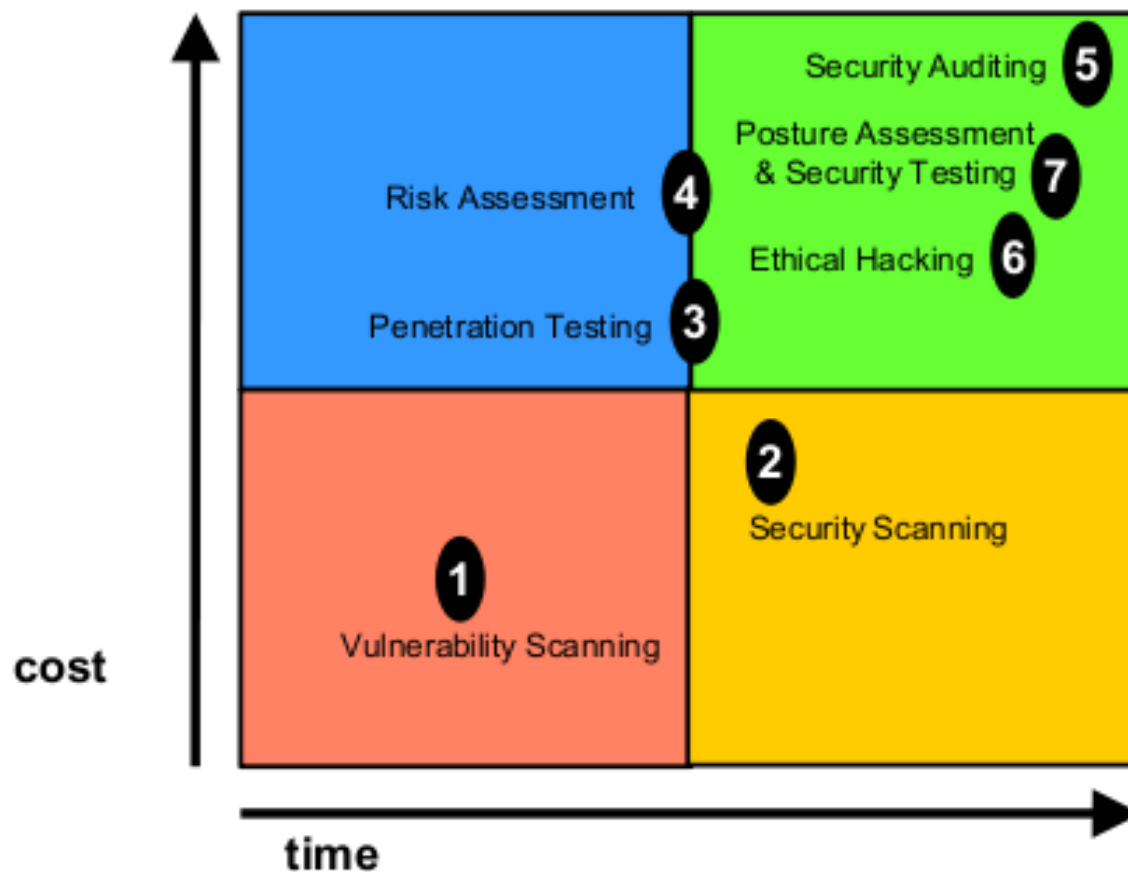
- Scary out there
  - Customized malware!
    - Zeus Trojan & ACH Transfers
      - Read Brian Krebs if you're not already
  - Verizon Data Breach report
    - Qualys, Veracode, etc, etc, etc
  - SQL Injection Worms, Massive Botnets

But ... most people don't know it, or have too much to do already

# Security Assessment Today

- Everybody needs it – usually because of regulation, sometimes because of best practice
  - Verticals:
    - Financial
    - Government (Federal, State, Regional)
    - Healthcare, Retail, Education, Insurance, etc
- Who does it
  - Lots of local firms, boutique shops
  - Government Contractors
  - Big4

# What Kind of Security Assessment?



# Penetration Testing Today

- Business Side
  - SOWS / RFPs,
  - Generally Small Teams
- Technical Side
  - Network / Application (Web / Fat Client)
  - Wireless
  - Tons of specialties, but jack-of-all-trades is helpful

**Best YouTube video ever ...**



**in...3...2...1...**

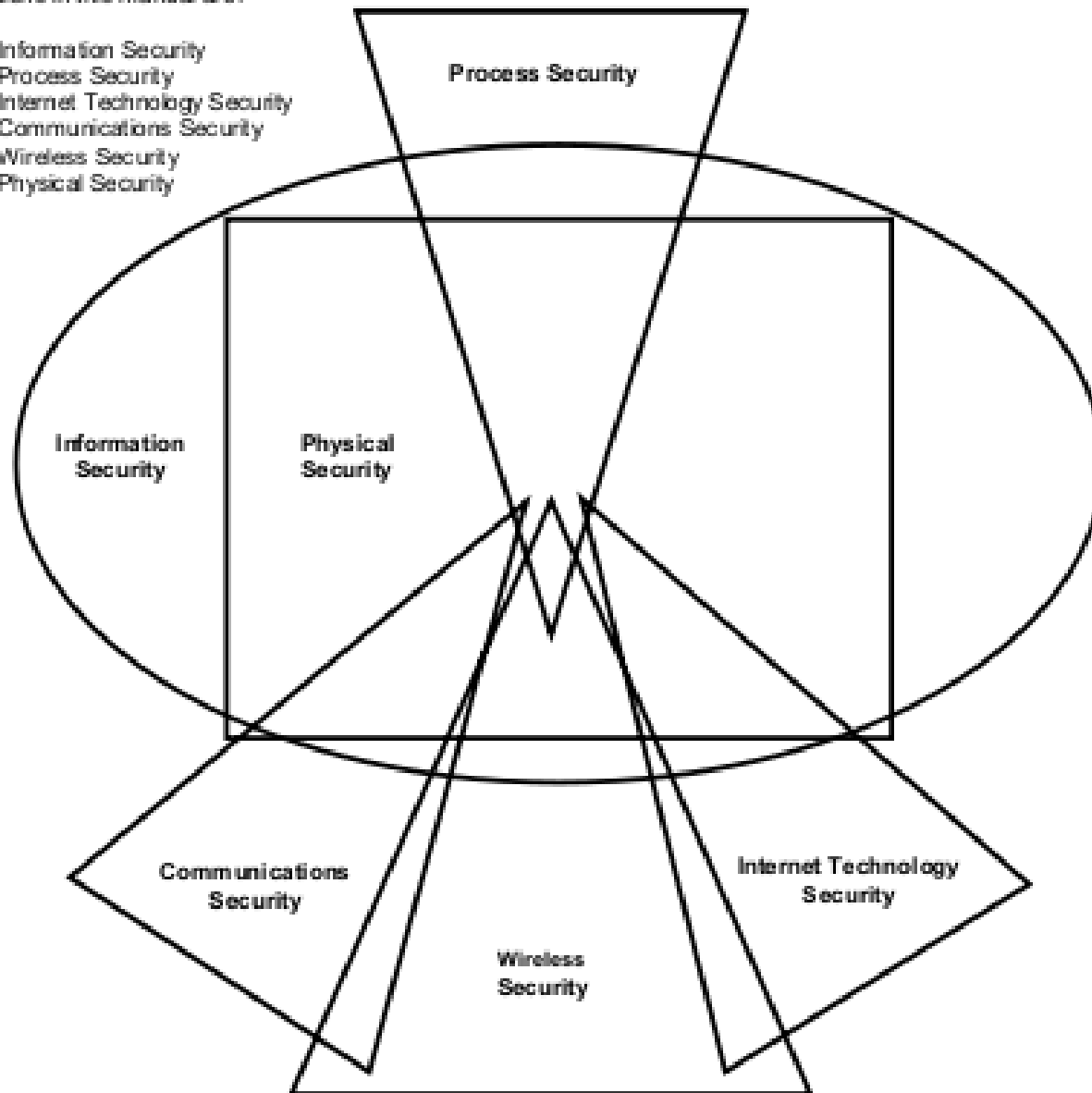


# Methodology

- Hacking Exposed 5-step Model
  - recon,discover, enumerate, exploit, persist
- Vulnerability-Assessment.co.uk
- NIST SP800-115 & ISSAF
- OWASP for web
- OSSTMM
  - Much more than just infosec – “domains” of security
  - Measurement

The sections in this manual are:

1. Information Security
2. Process Security
3. Internet Technology Security
4. Communications Security
5. Wireless Security
6. Physical Security



# Penetration Testing Today

- Focus on Network / Application
- Remote Engagements
  - Mostly App layer
    - Asp.net, J2EE, various enterprise frameworks
    - Asp,jsp,php (win!)
  - Or Social Engineering
  - “Harder than it used to be” – but get a shell, and you’re internal

# Penetration Testing Today

- Internal Engagements
  - Some physical work
  - Still wide open
  - Certain vulnerabilities still not well understood & relatively easy to exploit
    - Arp spoofing / mitm
    - Windows exploitation
    - Wireless is getting better
  - Generally some weak points

**Helping othurs...**

**it's hows we roll...**

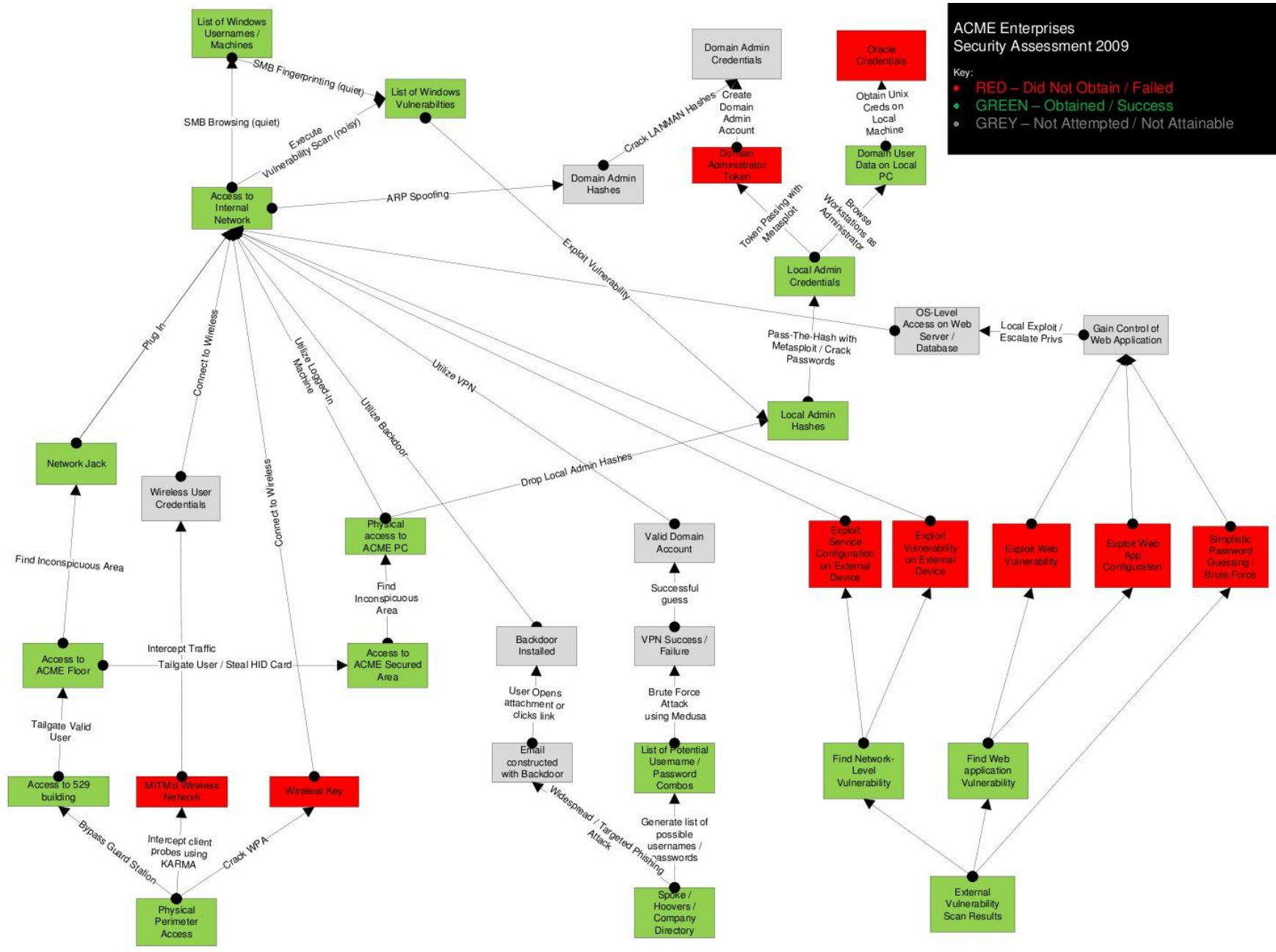
# Penetration Testing Today

- Typical engagement – 1-2 weeks
- Once / year, mandated by various regs
- Lots of silly constraints
  - “Don’t touch these systems, we know they’ll break”
  - Social Engineering not allowed
- Working with the security team
  - Which is generally part of IT
- Generate a report w/ remediation

# ACME Enterprises Security Assessment 2009

Key:

- RED – Did Not Obtain / Failed
- GREEN – Obtained / Success
- GREY – Not Attempted / Not Attainable



# Toolkit

- Technically, Penetration testing requires:
  - An ability to learn the environment & improvise
  - Bag of tricks
    - Lots of tools / techniques.
      - If you don't mind being noisy, start with a vulnscan
      - If an admin uses it, so can you :)
    - Practical attacks!
- Commercial Tools
  - Handy if you're short on time / patience!



**FORR THE LUVS OF GODZ!**



**NO MOR WATURRRRRR!!!!!!**

# What is Metasploit?

- Metasploit Project
  - HD Moore & Friends
  - A community-driven project since 2003
- Rapid7
  - Maker of NeXpose Vulnerability Management
  - Purchase the Project from HD
- Metasploit Framework
  - The original open-source exploit framework

# Metasploit Versions

- 1.0 released in 2003 (Perl)
- 2.0 released in 2004 (Better Perl)
  - 2.7 released in late 2006
- 3.0 released in 2007 (Ruby)
  - 3.2 released in late 2008
  - 3.3 released in late 2009
  - 3.4 released in early 2010
  - 3.4.1 is the most recent release
    - Metasploit now has 567 exploits and 283 auxiliary modules (up from 551 and 261 in v3.4)

# Metasploit Filesystem

- Organized by directory
  - lib: the meat of the framework code base
  - data: editable files used by Metasploit
  - tools: various useful command-line utils
  - modules: the actual modules
  - plugins: loadable plugins
  - scripts: metepreter and other scripts
  - external: source code and third-party libs

# Metasploit Libraries

- Rex is the basic library for most tasks
- Sockets, protocols, text transformations
- SSL, SMB, HTTP, XOR, Base64, Unicode
- Msf::Core provides the 'basic' API
  - Defines the framework
- Msf::Base provides the 'friendly' API
  - Simplified APIs for the framework

# Metasploit User Interfaces

- msfconsole
  - This is what you should ALWAYS use
  - Most features and the most stable
  - Windows supported via Cygwin
- msfgui, msfweb, msfcli
  - Useful for specific tasks
  - Less supported

# Metasploit Core Concepts

- Exploits
  - Defined as modules which use payloads
  - Exploits without payloads: Auxiliary
- Payloads, Encoders, Nops
  - Payloads run remotely
  - Encoders make sure they get there
  - Nops keep payloads sizes consistent

# Demo Strategy

Show you cool stuff, and where to learn more  
5 min / demo



# Quick & Dirty Reconnaissance

Propecia.c

Vulnerability scanner

Metasploit Aux Modules

# Quick & Dirty Network Attacks

Metasploit MS08\_067 DEMO – msfconsole

Metasploit MS08\_067 DEMO – msfgui

Autopwn DEMO

# Epic Tangent: Meterpreter Scripting

- Start a process

```
note = client.sys.process.execute('notepad.exe', nil, {'Hidden' => true })
```

- Get all process

```
processes = client.sys.process.get_processes.sort_by { |ent| ent['pid'] }
```

- Migrate

```
client.core.migrate(target_pid)  
server = client.sys.process.open
```

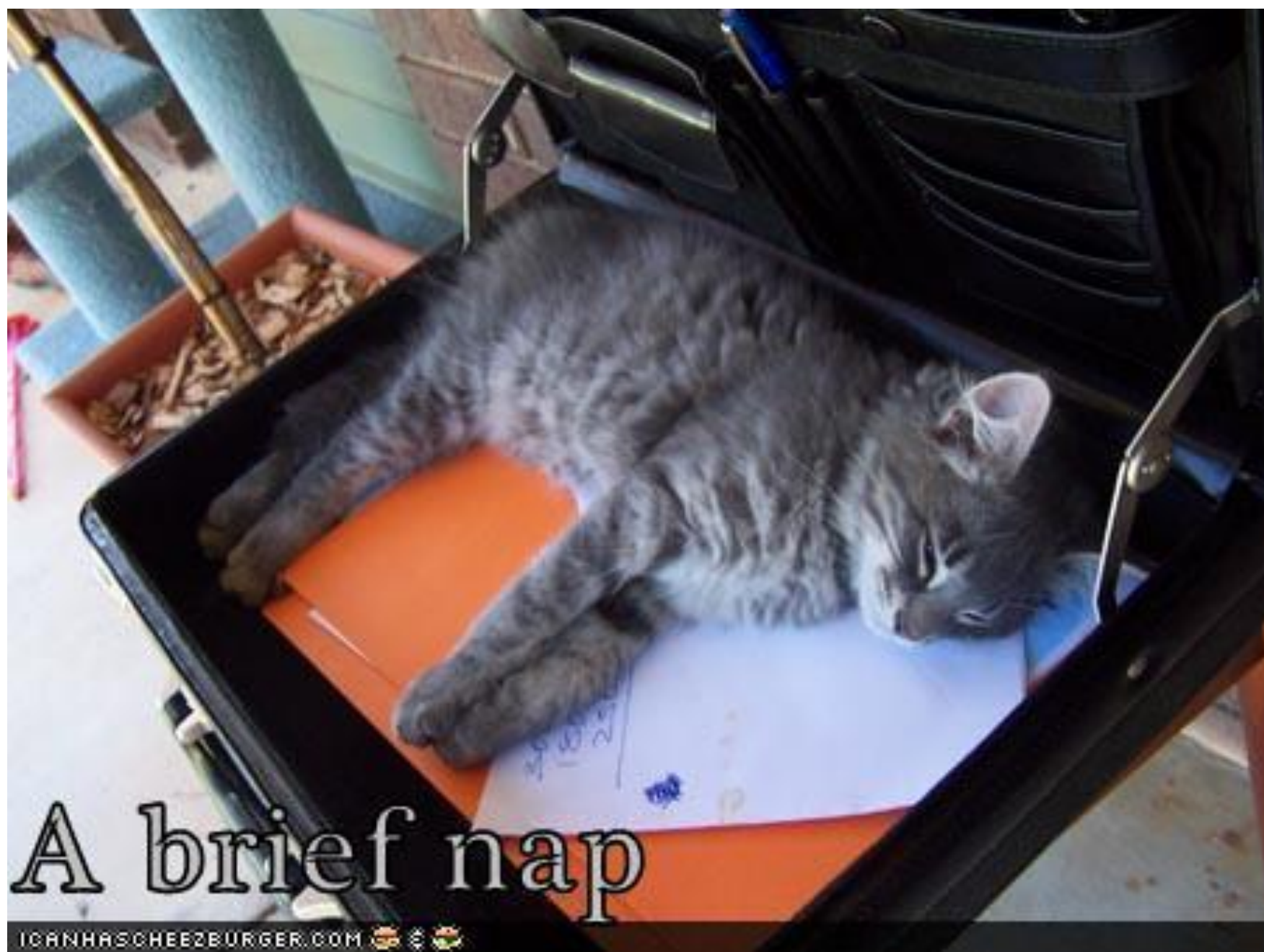
- Kill a bunch of processes

```
annoying = ['notepad.exe', 'calc.exe']  
client.sys.process.get_processes().each do |x|  
  if (annoying.index(x['name'].downcase))  
    print_status("Killing off #{x['name']}...")  
    client.sys.process.kill(x['pid'])  
  end  
end  
end
```

# Quick & Dirty Physical Attacks

Kon-Boot DEMO

Hash-Stealing DEMO



A brief nap

# Quick & Dirty Windows Attacks

DLL Hijacking DEMO (maybe)

Pass-The-Hash DEMO

# Quick & Dirty Social Engineering

Browser AutoPWN DEMO

Meterpreter Connect-Back DEMO

SET DEMO

All your base are belong to us!





# Quick & Dirty Web Hacking

Burp DEMO

# Scripting For Fun & Profit

- PacketFu Demo

Le Questions?

# Network Pentesting Resources

- <http://www.backtrack-linux.org/> - Backtrack Linux
- [http://www.owasp.org/index.php/Category:OWASP\\_Testing\\_Project](http://www.owasp.org/index.php/Category:OWASP_Testing_Project) OWASP Testing Guide
- <http://www.isecom.org/osstmm/> OSSTMM
- <http://www.vulnerabilityassessment.co.uk/> VulnAssessment.co.uk Methodology
- <http://www.exploit-db.com/> Exploit DB – Daily updates for exploits
- <http://seclists.org/fulldisclosure/> Pretty much every bug hits this list at some point

# Metasploit Resources

- <http://www.metasploit.com/>
  - <http://blog.metasploit.com/> (Blog, News)
  - <http://www.metasploit.com/redmine/projects/framework/activity> (RSS / Tracker / New Features)
- <http://www.offensive-security.com/metasploit-unleashed/> (metasploit tutorial)
- <http://blog.metasploit.com/2010/05/introducing-metasploitable.html>  
(metasploitable – exploitable VM)

# Reading List

- Basics
  - CounterHack Reloaded
  - Hacking Exposed
  - The Art of Software Security Assessment
  - \$SCRIPTING\_LANGUAGE book
  - Web Application Hacker's Handbook
- Advanced Reading
  - Jon Erickson's Hacking: The Art of Exploitation
  - Shellcoders Handbook