How To Prevent Rolling Spam Factories

Ryan Ware

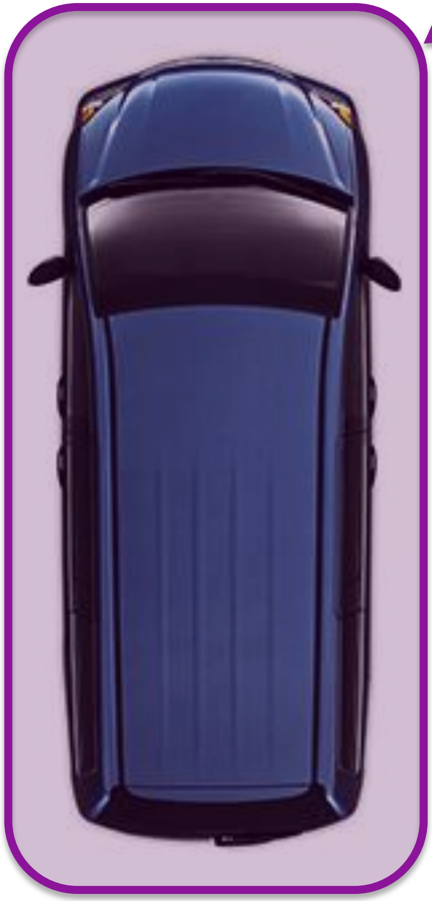
Lead Security Architect

INTEL OPEN SOURCE TECHNOLOGY CENTER

The Onslaught Is Coming

Connectivity Evolution

Threat Space



No IP Connectivity

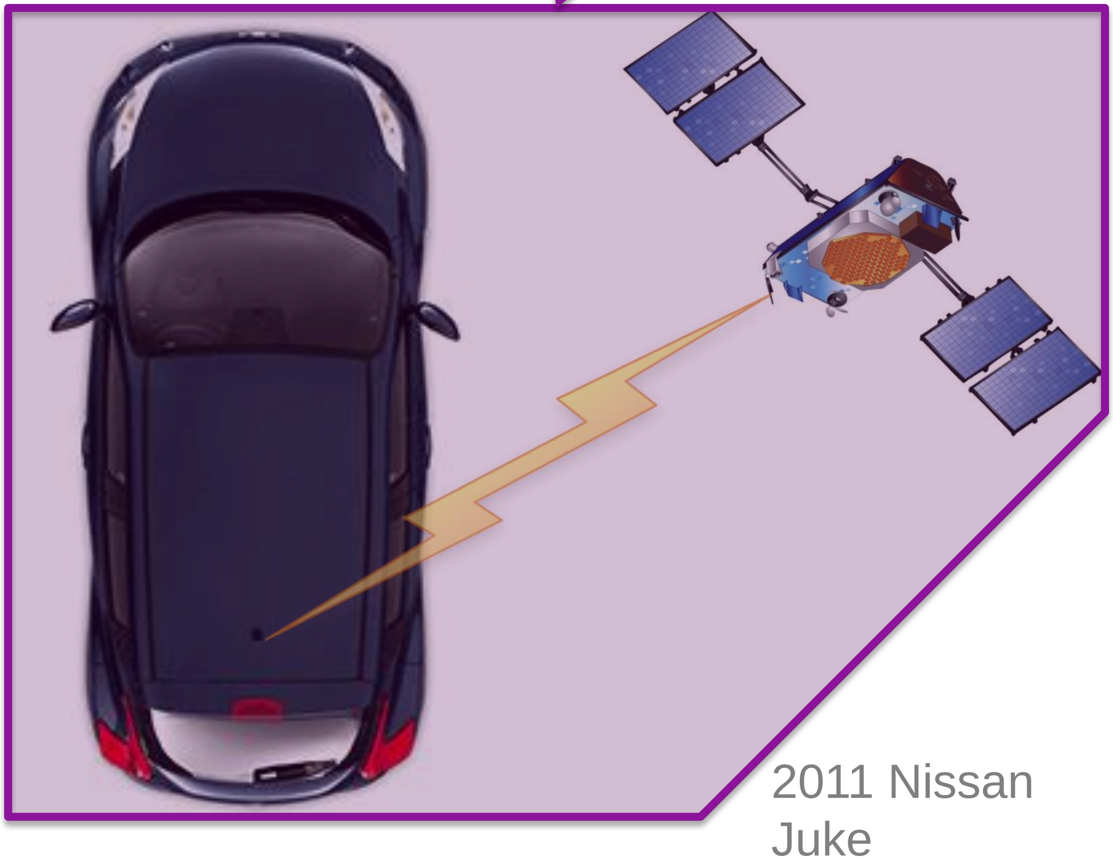
- A/V Input
- CD
- DVD
- Analog Radio

2002 Honda Odyssey

Completely Self Contained

Connectivity Evolution

Threat Space



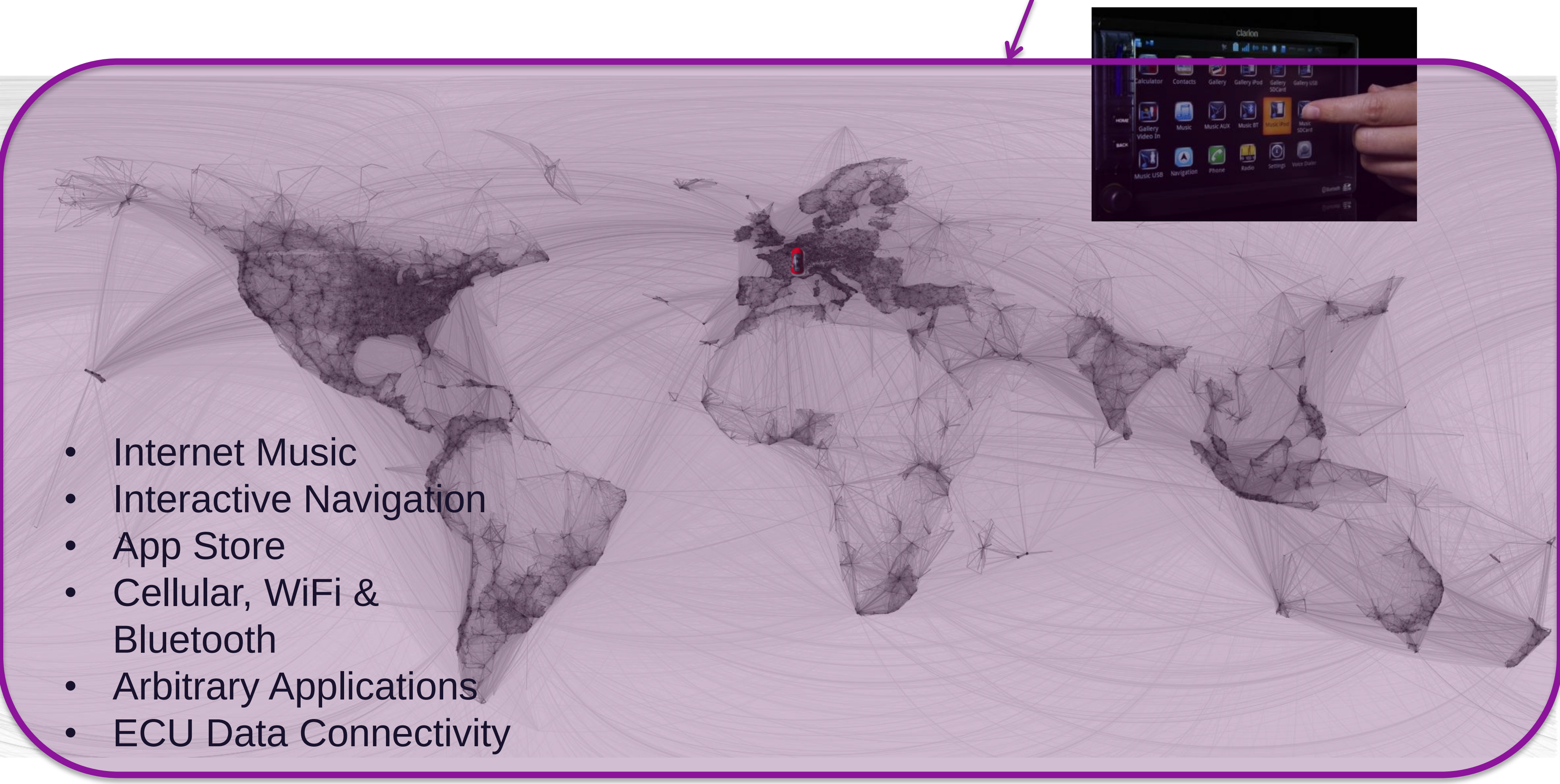
2011 Nissan Juke

- Digital Music
- Traffic Data
- A/V Input
- USB
- Bluetooth
- CD
- Analog Radio

Mostly Self Contained

Connectivity Evolution

Threat Space

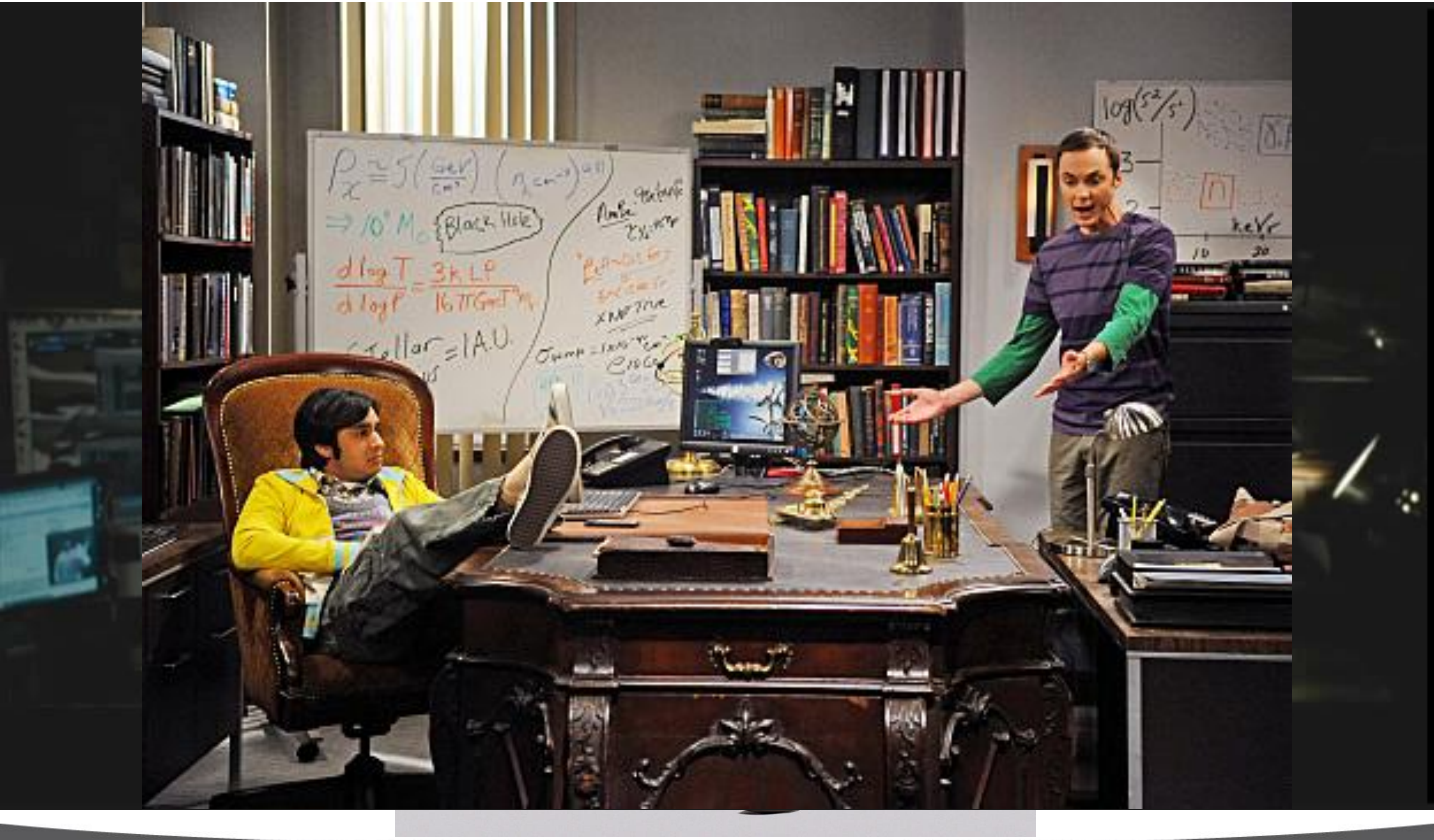


- Internet Music
- Interactive Navigation
- App Store
- Cellular, WiFi & Bluetooth
- Arbitrary Applications
- ECU Data Connectivity

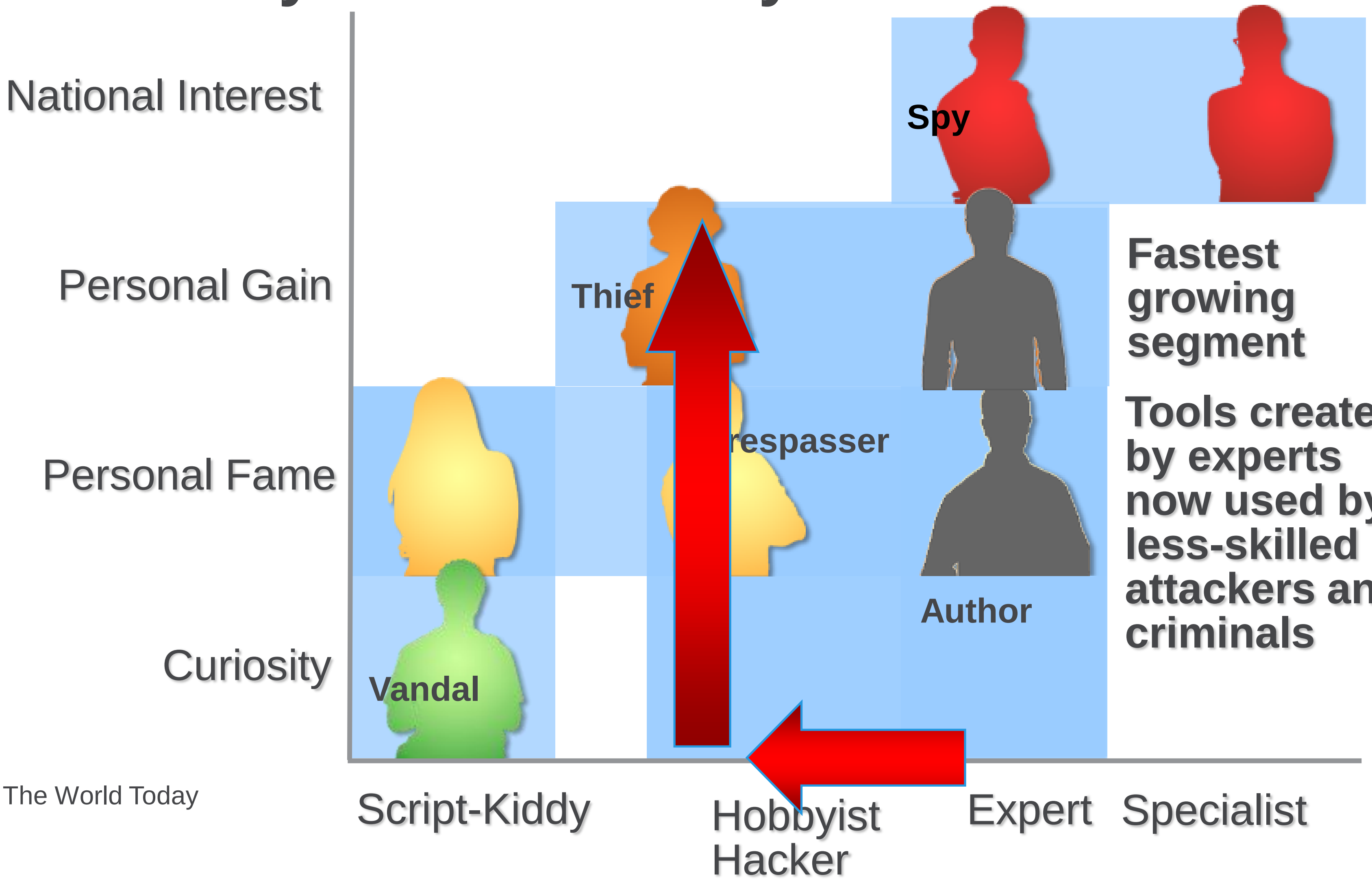
Absolutely No Containment

Who Are We Protecting From?

Not Your Mother’s Hacker



Security Hackers Ecosystem



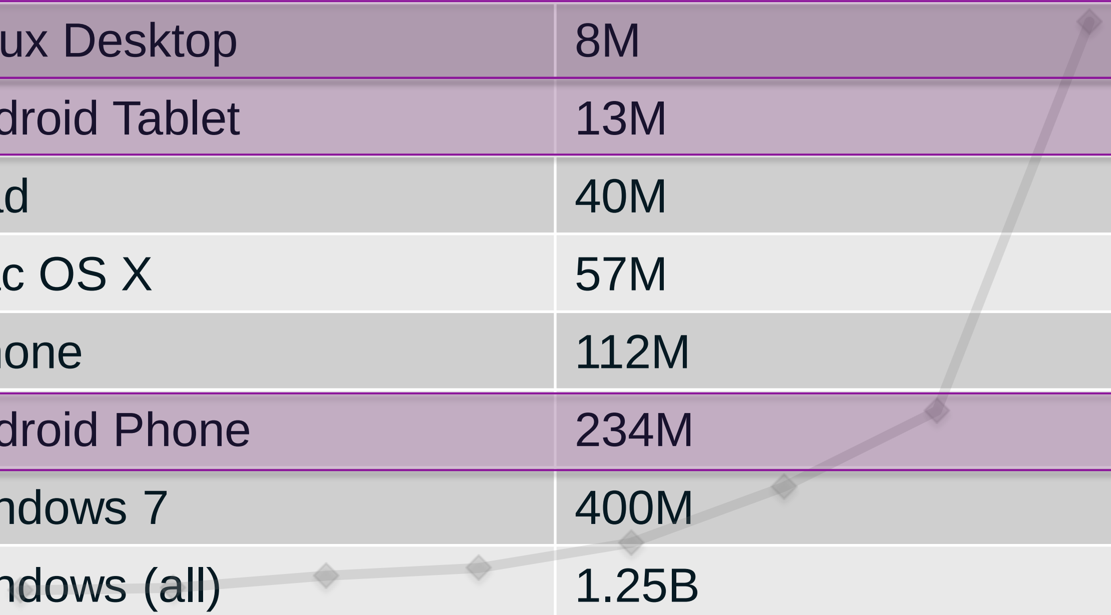
Fastest growing segment

Tools created by experts now used by less-skilled attackers and criminals

The World Today

Why Has Windows Been The Largest Target?

Approximate Install Base	EOY 2011
Linux Desktop	8M
Android Tablet	13M
iPad	40M
Mac OS X	57M
iPhone	112M
Android Phone	234M
Windows 7	400M
Windows (all)	1.25B

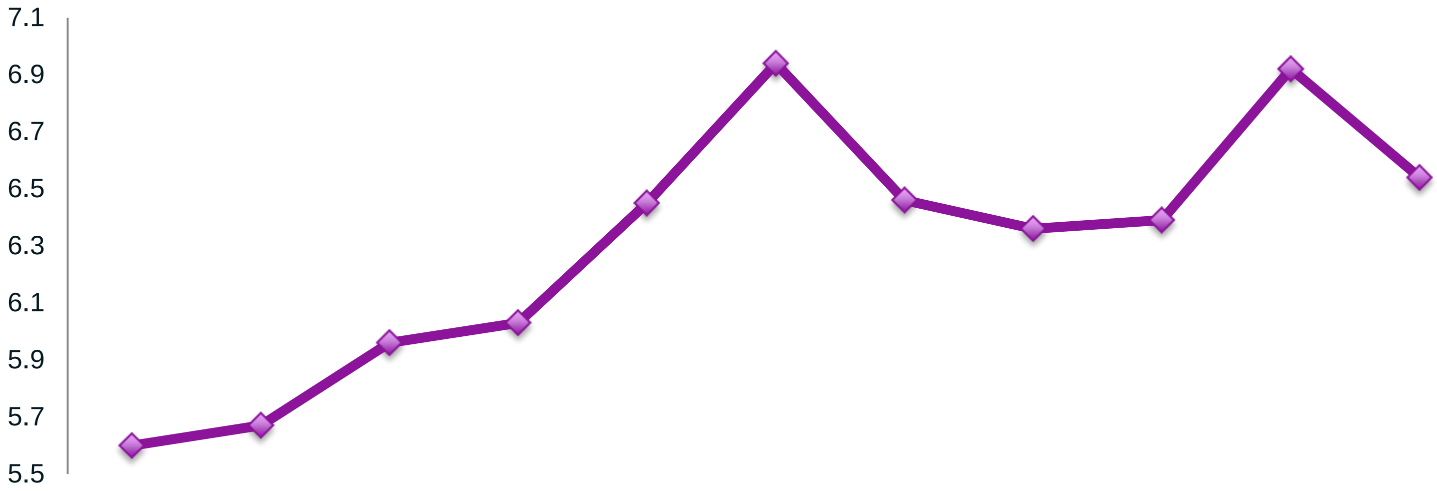


As a software developer trying to make money, what is your choice?

Choice for malware developer is same as for any other kind of developer!

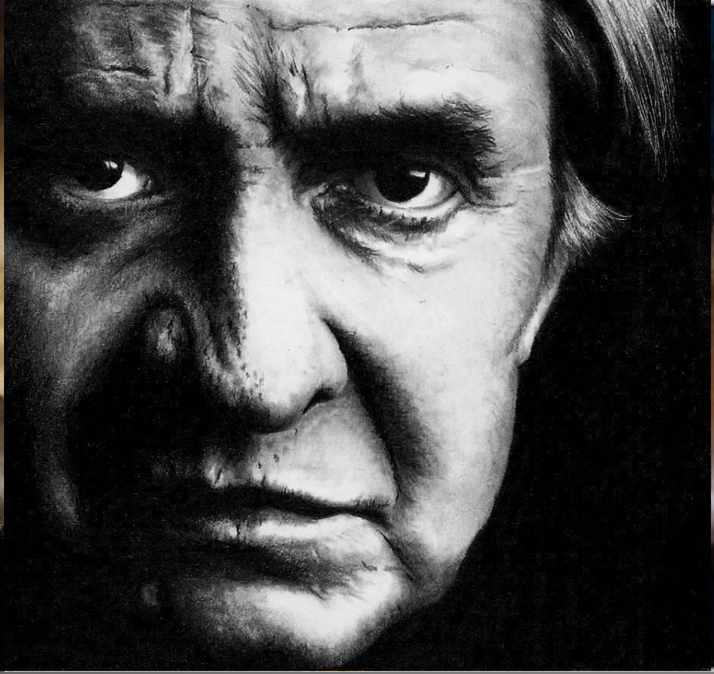
One of the primary ways a malware author makes money is to create a botnet and then lease it to spammers

Then Why Mac OS X Flashback?



- *Some* market share increase.
  - Increasing numbers **does** mean potential for increased money
- However, **primary** motivation is fame.
  - Apple's $70B brand name generates news
  - Currently 2,640,000 Google hits for "Mac OS X Flashback malware trojan"
- Apple has shipped their 3rd security update for this issue

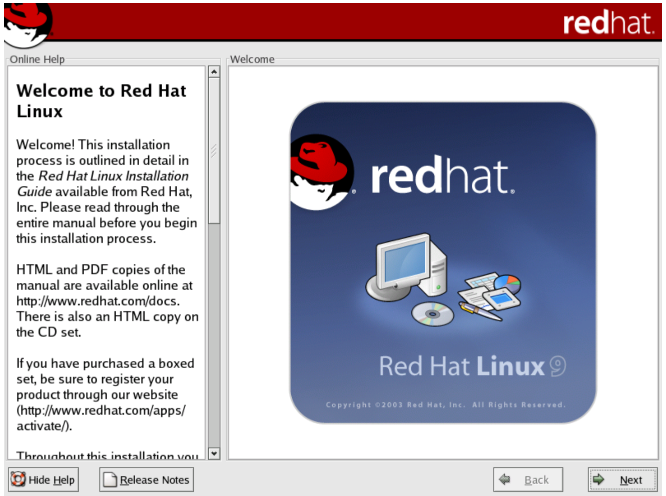
All this over 600,000 infected Macs; 1.05% of estimated install base



Magical Year of 2003

Red Hat Linux 9

- Kernel 2.4.20
  - POSIX Thread Library (NPTL)
  - Beta ACPI Support
  - Access Control Lists & Extended Attributes
  - User Mode Linux
- XFree86 4.3.0
  - Pre Xorg
- GNOME 2.2
  - 18 Major releases old
- KDE 3.1
  - 3.x codeline defunct
- Mozilla 1.2.1
  - Pre Firefox
- OpenSSL 0.9.7a
  - 12 versions old
  - No 0.9.7x updates since 2006



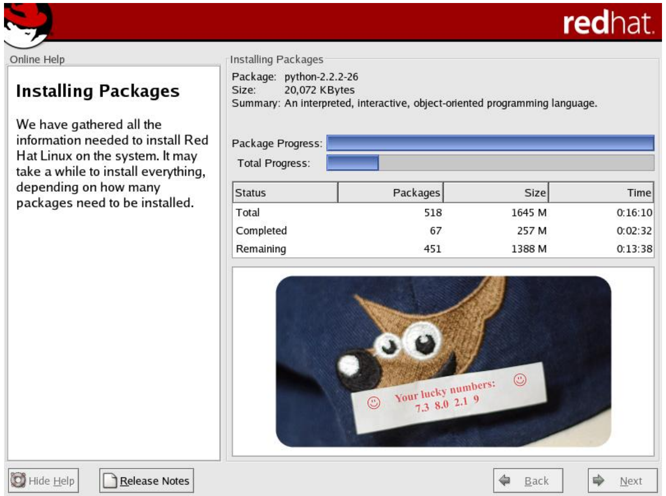
Welcome to Red Hat Linux

Welcome! This installation process is outlined in detail in the *Red Hat Linux Installation Guide* available from Red Hat, Inc. Please read through the entire manual before you begin this installation process.

HTML and PDF copies of the manual are available online at http://www.redhat.com/docs. There is also an HTML copy on the CD set.

If you have purchased a boxed set, be sure to register your product through our website (http://www.redhat.com/apps/activate/).

Throughout this installation you



Installing Packages

We have gathered all the information needed to install Red Hat Linux on the system. It may take a while to install everything, depending on how many packages need to be installed.

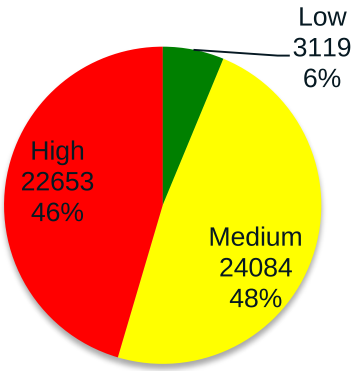
Package: python-2.2.2-26
Size: 20,072 KBytes
Summary: An interpreted, interactive, object-oriented programming language.

Package Progress:
Total Progress:

Status	Packages	Size	Time
Total	518	1645 M	0:16:10
Completed	67	257 M	0:02:32
Remaining	451	1388 M	0:13:38

What Are These CVE, CVSS and CWE Things?

- Common Vulnerability & Exposures (CVE)
  - Started in 1999 by MITRE Corporation
  - Keeps a database of all *publicly* known computer security defects
  - Since 1999: 49,856 registered security defects
    - Average of 7 vulnerabilities per day
- Common Vulnerability Scoring System (CVSS)
  - Industry standard method of assessing severity of computer security vulnerabilities based on measurement and expert assessment.
  - Widely Adopted
    - 0.0 – 3.9: Low severity
    - 4.0 – 6.9: Medium severity
    - 7.0 – 10.0: High severity



- Common Weakness Enumeration (CWE)
  - Sponsored by MITRE
  - Formal list of software weakness types
    - Buffer Overflows, Structure & Validity Problems, Channel & Path Errors, etc.

Wheel of Linux Kernel Fortune!!!

Number of Vulnerabilities:

2003:	19
2004:	51
2005:	133
2006:	90
2007:	63
2008:	70
2009:	105
2010:	124
2011:	83

Average 1.53 CVE/Week Over 9 Years

Linux Kernel Details

Year	# of Vuln	DoS	Code Execution	Overflow	Memory Corruption	Bypass Something	Gain Information	Gain Privilege
2003	19	8		2		1	3	4
2004	51	20	5	12			5	12
2005	133	90	19	19	1	6	5	7
2006	90	61	5	7	7	5	3	3
2007	63	41	2	8		3	7	7
2008	70	44	3	17	4	4	6	10
2009	105	66	2	22	7	8	11	22
2010	124	67	3	16	7	8	30	14
2011	83	62	1	21	10	1	21	9
Total	738	459	40	124	36	36	91	88

Critical Linux Kernel Defects (CVEE Scores 8-10)

- **CVE-2003-0959:** Denial of service or root privilege escalation
  - Multiple overflows in 32bit emulation
- **CVE-2004-1017:** Arbitrary code execution
  - Multiple overflows in io_edgeport driver
- **CVE-2004-1137:** Denial of service or arbitrary code execution
  - Multiple vulnerabilities in Internet Group Management Protocol (IGMP) handling
- **CVE-2006-1368:** Denial of service
  - Buffer overflow in USB Gadget RNDIS implementation
- **CVE-2006-1523:** Arbitrary code execution
  - Vulnerability in __group_complete_signal function in RCU signal handling
- **CVE-2006-1857:** Denial of service or arbitrary code execution
  - Buffer overflow in Stream Control Transmission Protocol (SCTP)
- **CVE-2006-6535:** Denial of service
  - dev_queue_xmit function can fail before calling local_bh_disable
- **CVE-2008-1673:** Denial of service or arbitrary code execution
  - Multiple vulnerabilities in asn1 implementation
- **CVE-2008-3496:** Arbitrary code execution
  - Buffer overflow in format descriptor parsing in uvc_parse_format

Critical Linux Kernel Defects (CVEE Scores 8-10)

- **CVE-2008-3915:** Arbitrary code execution
  - Buffer overflow in NFSv4 implementation
- **CVE-2008-4395:** Arbitrary code execution
  - Multiple buffer overflows in ndiswrapper module
- **CVE-2008-5134:** Denial of service
  - Buffer overflow in wireless driver function lbs_process_bss
- **CVE-2009-0065:** Denial of service
  - Buffer overflow in Stream Control Transmission Protocol (SCTP)
- **CVE-2009-4538:** Denial of service
  - Improper checks in e1000e driver
- **CVE-2010-2495:** Denial of service or arbitrary code execution
  - Improper check in L2TP driver
- **CVE-2010-2521:** Denial of service or arbitrary code execution
  - Multiple buffer overflows in XDR implementation of NFS server
- **CVE-2010-3705:** Denial of service
  - Improper validation of hmac_ids array of an SCTP peer
- **CVE-2011-2497:** Denial of service or arbitrary code execution
  - Integer underflow in Bluetooth driver

This Was Just The Linux Kernel!

Only **1** of the packages for our hypothetical, 2003 era system!

How To Protect Ourselves

Reduce Attackable Surface Area

- Do not include unused functionality!

- Install services only used by your product
- Expose only limited, well documented interfaces
- Remove Debug Interfaces
- There is no such thing as a “Private” interface

- Included functionality not used by the product is simply additional areas for a hacker to attack

Least Privilege

- Security Principle of only giving the needed privilege.
- It is of course easier to just run everything as root!
  - Even have less in the way if you just run in the kernel
- **When** your software fails, not running as root means you have additional protections
- Very little software needs to actually mmap() to entire physical address space

Use Compiler Defenses

- -Wformat -Wformat-security -Werror=format-security
  - Provides warnings about potential security holes.

```
trivial.c: In function 'main':
trivial.c:16: warning: format not a string literal and no format
arguments
```

- -fstack-protector-all
  - Provides canary based buffer overflow checks on the stack
  - Shuts application down with corrupt stack
- -D_FORTIFY_SOURCE=2
  - Silently replaces unbounded string function calls with bounded ones
  - Only done where gcc can determine the buffer size
- -fpic & -fpie
  - Generate position independent code for libraries (-fpic) and executables (-fpie).
  - Protects against “return to libc” attacks
- NX (XD) Bit
  - Utilize the No eXecute (or eXecute Disable) bit where possible.
  - 32-bit Intel needs PAE

Mandatory Access Controls

- Access control enforcement that supersedes Linux Discretionary Access Control (DAC)
- Many Different Linux Security Modules that enforce MAC
  - SELinux, Smack, TOMOYO, etc.
- Tizen and MeeGo have utilized Simplified Mandatory Access Control Kernel (Smack)
  - SELinux Reference Policy more than 900,000 lines
  - SEAndroid Reference Policy is one quarter in size, but pressure already trying to grow it
  - Smack allows creation of an arbitrary number of security domains keeping things simple unless necessary
  - Example: 32 different security domains with 100 lines of Smack rules
- Smack support recently added to:
  - Dbus
  - Xorg
  - udev

Systemd & Cgroups

- Systemd is replacement for System V init daemon providing more efficient startup and process control
- Cgroups allows resource limits on processes or process groups
- Systemd automatically places system services and groups of services into appropriate cgroups

- Ensures misbehaving processes can be controlled and shut down
- Potentially include Linux Containers-like features

Linux Containers (LxC)

- Userland Application to configure Linux Kernel isolation features
- **Not** a virtualization solution
  - **Does not protect from user to kernel privilege escalation attacks**
- Allows isolating processes in different ways:
  - Process Namespace: Isolate processes from seeing one another
  - IPC Namespace: Ensures processes cannot share IPC’s
  - UID Namespace: Separate UID tables
  - Network Namespace: Separate network devices
  - UTSName Namespace: Separate host names
  - Mount Namespace: Separate mounted devices
- Can be combined with filesystem snapshots (btrfs) to create disposable environments

Integrity

- Integrity Measurement Architecture
  - Provides runtime filesystem integrity
  - Keeps hash of all files in kernel memory
  - Kernel verifies hash on file open
- Extended Verification Module
  - Provides offline filesystem integrity
  - Keeps HMAC of file extended attributes and hash
  - Depends on HW security subsystem (TPM)
- DM-Integrity
  - Addition to DM-Crypt
  - Extends to use HMAC over files
  - Depends on HW security subsystem

Summary

- The environment we're deploying products into is not changing for the better
- Supporting a product lifetime of 10 years provides unique difficulties
- Many different technologies and techniques to improve our protection:
  - Attack Surface Reduction
  - Least Privilege
  - Mandatory Access Controls
  - Systemd
  - Cgroups
  - Linux Containers
  - Integrity

INTEL OPEN SOURCE
TECHNOLOGY CENTER