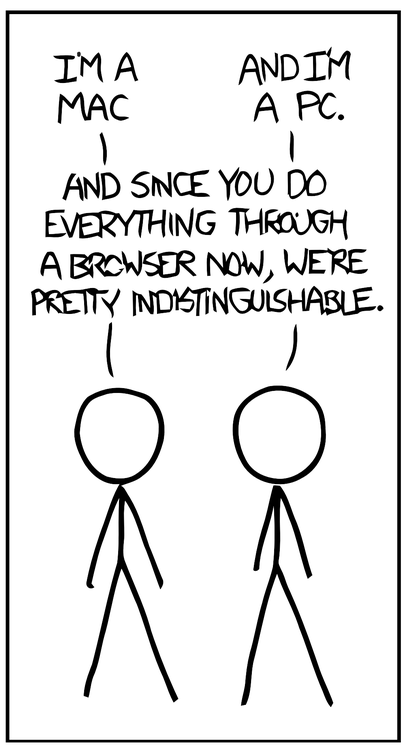
Cross-Platform Malware: Write Once, Infect Everywhere

Martina Lindorfer, Matthias Neumayr, Juan Caballero, Christian Platzer
Vienna University of Technology, IMDEA Software Institute

Introduction

- Programmers aim at writing a program once and then using it on different computing platforms: **„Write once, run everywhere“**
- Benefits include code reuse, reduced development time and easier maintenance.
- This paradigm is extended in benign software, but it is not yet prevalent in malware: The majority targets Windows with Android Malware recently growing.
- Supporting a new platform boils down to a cost-genefit analysis: the income from supporting new platforms vs. the additional investment in software development and distribution.
- A cost-effective way of distributing malware is through drive-by downloads leveraging exploits for X-platform vulnerabilities.

In this work we explore:
- X-platform malware and how it achieves portability
- X-platform vulnerabilities and their availability in commercial exploit kits



Source: http://xkcd.com/934/

Overview

A X-platform program is a program that is portable across different OS families.

Programs become portable ...
- using programming languages compiling to bytecode e.g. Java, .NET
- on source code level using standardized interfaces e.g. POSIX
- on source code using interpreted languages e.g. Perl, Python
- running on top of other X-platform programs e.g. web browsers, office applications

A X-platform vulnerability is a software defect present in platform-independent code of a X-platform program.

TECHNISCHE UNIVERSITÄT WIEN
Vienna University of Technology

Contact:
mlindorfer@iseclab.org
mneumayr@iseclab.org
juan.caballero@imdea.org
cplatzer@iseclab.org

X-Platform Malware

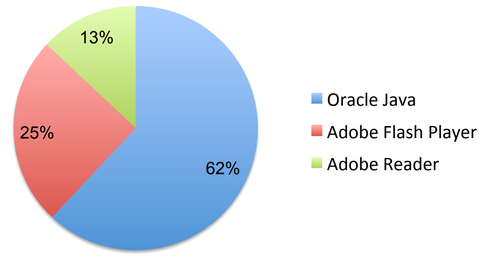
- We found 14 X-platform malware families already observed in the wild.
- We found three proof-of-concept malware samples.
- Four families use exploits to get installed on the target hosts, while the remaining nine rely on social engineering (SE) for distribution.
- X-platform malware is distributed as
  - Source code: Python, JavaScript, Perl, Ruby
  - Binary code: PE, ELF, MACH-O
  - Bytecode: Java, .NET

Family	Date	Distribution	Platforms
Badbunny	07/09	SE	JavaScript Perl Ruby
Boonana	10/10	SE	Java Java
ZitMo	09/10	SE	Java Java Java* Java*
Olyx	06/11	Exploit	PE MACH-O
Tibet	03/12	Exploit	PE MACH-O
Flsplysc	04/12	Exploit	PE Python
Crisis	04/12	SE	PE MACH-O PE*
LilyJade	05/12	Exploit	JavaScript JavaScript JavaScript
GetShell	07/12	SE	Java Java Java
Netweirdrc	08/12	SE	PE ELF MACH-O
jRAT	10/12	SE	Java Java Java
Ssucl	01/13	SE	PE Java
MinecraftHack	03/13	SE	Java Java
Janicab	07/13	Exploit/SE	VB Script Python
Clt10 (PoC)	??/06	-	ASM ASM
Yakizake (PoC)	08/07	-	.NET .NET
Clapzok (PoC)	05/13	-	ASM ASM ASM

X-platform malware, earliest date reported and supported platforms (* mark platforms supported later)

X-Platform Vulnerabilities

- X-platform vulnerabilities exist in
  - Browser plugins (Java, PDF, Flash)
  - Web browsers (Firefox, WebKit)
  - Desktop applications (Microsoft Word)
- Java is by far the most vulnerable application



Most targeted software 2012 (Source: Kaspersky)

- Public exploits are available for almost all vulnerabilities. We verified their X-platform functionality with exploits from Metasploit.
- Commercial exploit kits contain exploits for most of the observed X-platform vulnerabilities.
- An attacker using drive-by download specialization services can already distribute malware to multiple platforms at essentially no extra cost.

CVE	Exp.	Kit	Metasploit Win XP	Metasploit Win 7	Metasploit Linux	Metasploit OS X
2009-0563	✓					
2009-3867	✓	✓	✓			
2010-3333	✓					
2011-1774	✓		✓			
2011-3544	✓	✓	✓	✓	✓	✓
2012-0507	✓	✓	✓	✓	✓	✓
2012-0779	✓	✓	✓			
2012-1723	✓	✓	✓	✓	✓	
2012-4681	✓	✓	✓	✓	✓	✓
2012-5076	✓	✓	✓	✓	✓	✓
2013-0422	✓	✓	✓	✓	✓	✓
2013-0431	✓	✓	✓	✓	✓	✓
2013-0640	Z		-	-	-	-
2013-0641	Z		-	-	-	-
2013-0758	✓		✓	✓	✓	✓
2013-1488	✓		✓	✓	✓	✓
2013-1491	Z		-	-	-	-
2013-2423	✓	✓	✓	✓	✓	✓

X-platform vulnerabilities, their availability in exploit kits and the verified functionality of Metasploit exploits (Z marks zero-day vulnerabilities)

Future Work

- Collect samples of the identified malware families
- Measure the amount of code reuse
- Collect exploits for the identified X-platform vulnerabilities and examine their X-platform capabilities
- Analyze X-platform exploits and malware in the wild through multi-platform honeyclients