Information & Communication Security (SS 15)

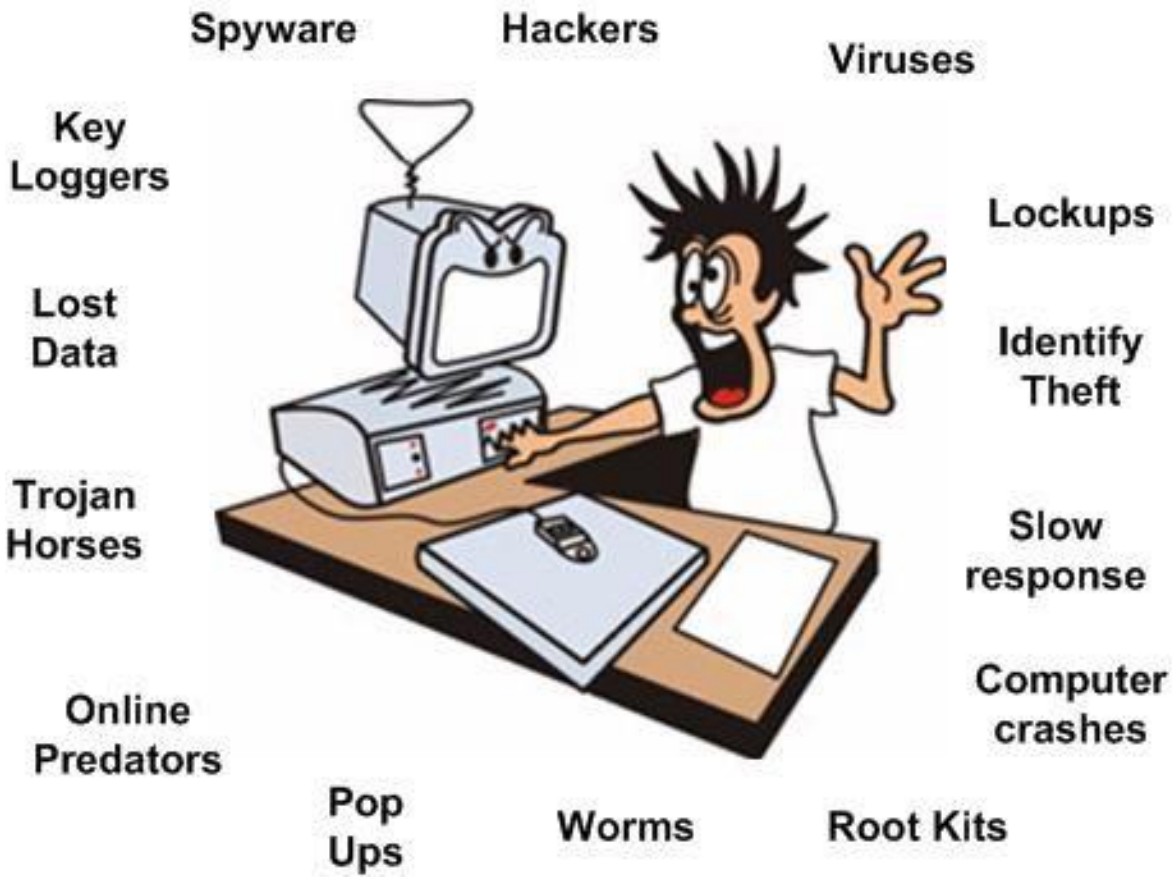
Computer Systems Security

Dr. Jetzabel Serna-Olvera
@sernaolverajm
Chair of Mobile Business & Multilateral Security
Goethe University Frankfurt
www.m-chair.de

Agenda

- Introduction
- Security Threats
- Operating System Security
- Mobile Malware
- Improving Security

Introduction



Source: ProactiveComputing.com

mobile business

Agenda

- Introduction
- Security Threats
  - Malicious Logic
  - Buffer Overflow
- Operating System Security
- Mobile Malware
- Improving Security

Malicious Logic

Definitions

- Is a set of instructions that cause a site's security policy to be violated.

  [M. Bishop, Introduction to Computer Security]

- A program implemented in hardware, firmware, or software, and whose purpose is to perform some unauthorized or harmful action.

  [ISO/IEC 2382-8]

- Hardware, software, or firmware capable of performing an unauthorized function on an information system.

  [National Information Systems Security (INFOSEC) Glossary 2000]

Malicious logic: known as *malicious code* or **Malware** (**Mal**icious soft**ware**).

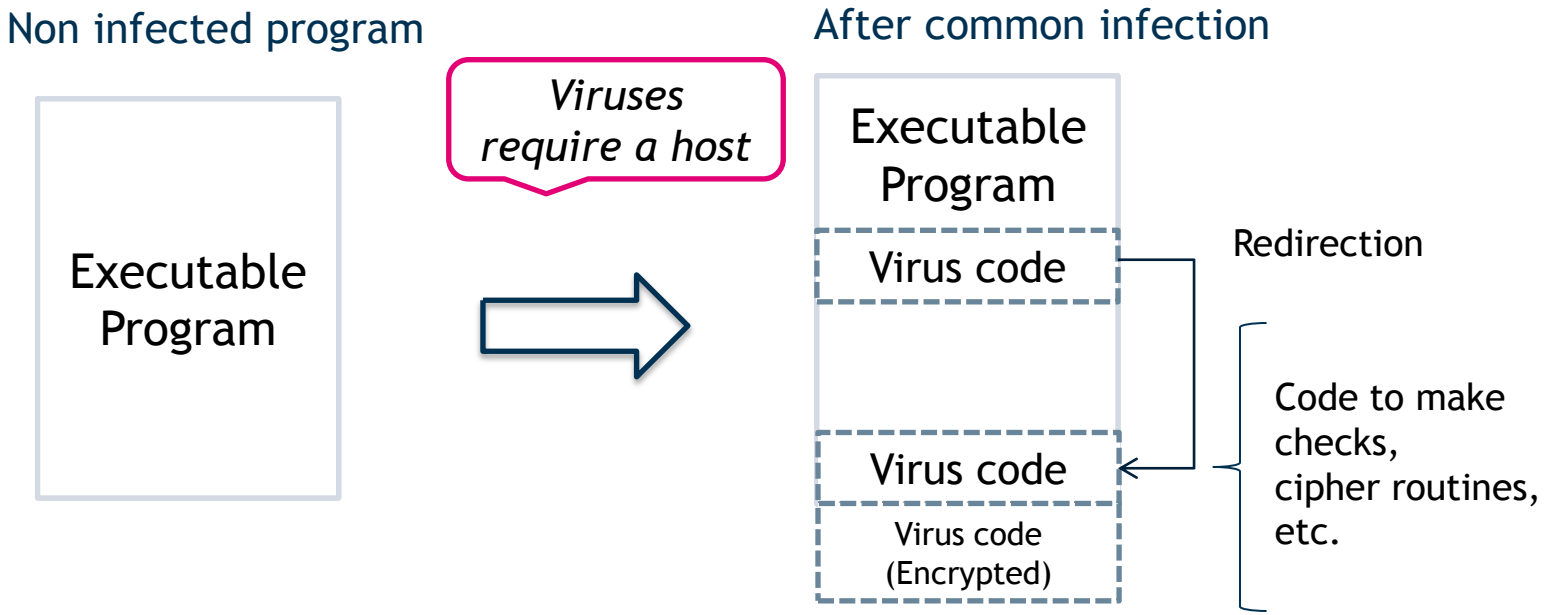
Types of malware

- Viruses
  - Self-spreading – it replicate relying on user activity
- Worms
  - Propagates autonomously from system to system, runs independently
- Trojan Horses
  - Programs with a covert purpose, non-spreading.
- Ransomware
  - Spreads like a normal computer worm: downloaded file or a network service vulnerability.

Viruses

Program that replicates itself, e.g., by inserting itself into one or more files, and that may perform some other action, too:

- *Insertion phase*: Virus is inserting itself into a file.
- *Execution phase:* Virus is performing some (possibly null) action.



Types of Viruses (1)

- Boot Sector Infector
  - Inserts itself into the boot sector of a disk
- Executable Infector
  - Infects executable programs, e.g. .EXE or .COM programs
  - May prepend itself (as shown) or put itself anywhere, fixing up binary so it is executed at some point
- Multipartite Virus
  - Can infect multiple platforms (e.g. either boot sectors or executables)
- TSR Virus (Terminate and Stay Resident)
  - Stays active in memory after the application is completed
- Stealth Virus
  - Conceals its presence on a system

Worms

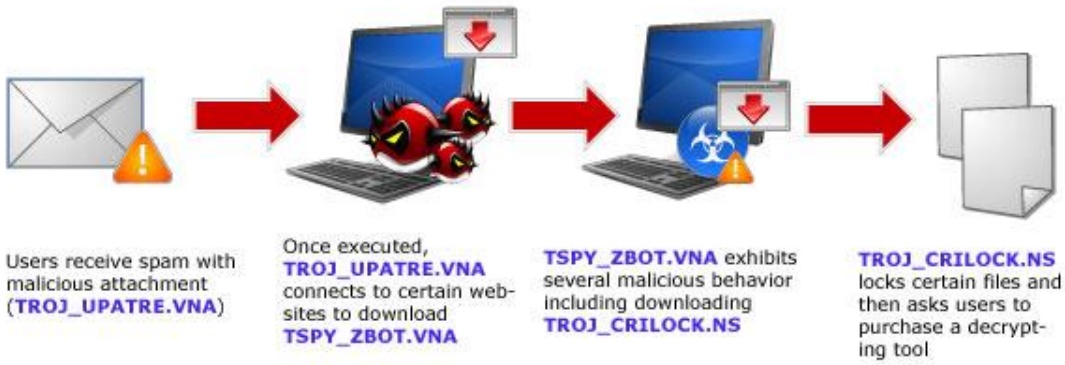
- A program that copies itself from one computer to another
- Origins: distributed computations
  - Animations, broadcast messages
- Segment
  - part of program copied onto workstation
  - processes data, communicates with worm's controller
  - Any activity on workstation causes segment to shut down.

Trojan Horses

- Program with an *overt* purpose (known to user) and a *covert* purpose (unknown to user)
  - Often called a Trojan
  - Named by Dan Edwards in Anderson Report *[Anderson72]*
- Example: NetBus
  - Designed for Windows NT system
  - Victim installs it:
    - Usually disguised as a game program, or integrated within one
  - Acts as a server, accepting and executing commands for remote administrator
    - This includes intercepting keystrokes and mouse motions and sending them to attacker.
    - Also allows attacker to upload, download files

Ransomware

It is a type of malware which restricts access to the computer system that it infects, and demands a ransom paid to the creator(s) of the malware in order for the restriction to be removed.



Example - ATM Attacks

- ATMs installed in non-branch areas with poor physical ATM security
- Low level of detection
- Malicious software physically installed per ATM
- Approximately 2.5hrs

Source: [http://www.bankinfosecurity.com/]

Source: [SecurityIntelligence: ATM Malware: The Next Generation of ATM Attacks]

Example - Carbanak

[VIDEO]

mobile business

Agenda

- Introduction
- Security Threats
  - Malicious Logic
  - Buffer Overflow
- Operating System Security
- Mobile Malware
- Improving Security

Buffer Overflow (1)

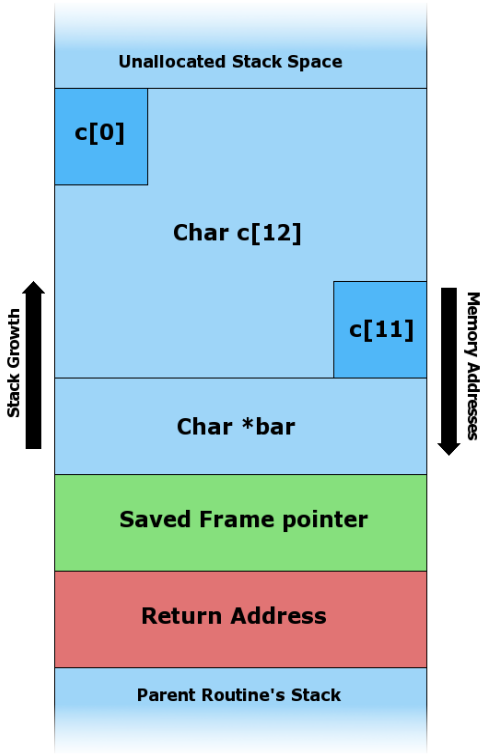
- Buffer overflow software bug
  - offer attackers the ability to write arbitrary data to memory: attempts to put more data in a buffer than it can hold
  - can overwrite a procedure return address
  - Languages affected: C, C++, Fortran, Assembly
- Persisted for decades
  - Users do not bother to install patches supplied (free) by software vendors.
- Simple example:

```
...
char buf[BUFSIZE];
gets(buf);
...
```

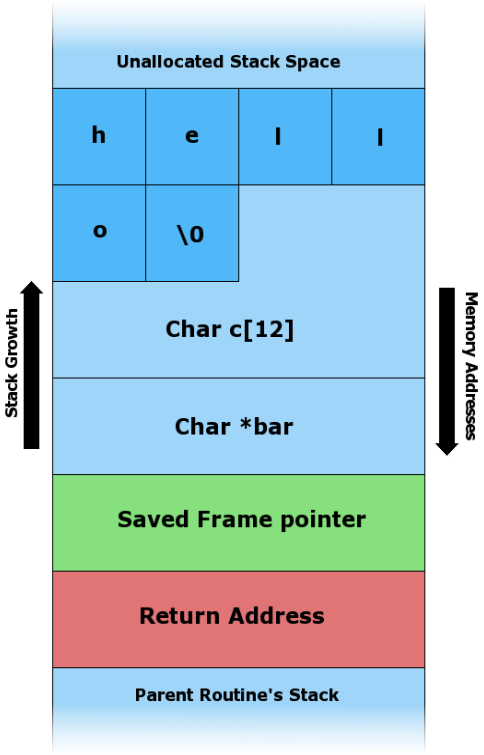
Buffer Overflow (2)

- Buffer overflow software bug
  - data larger than the variable allocated for it
  - can overwrite a procedure return address in the procedure call stack in memory
- Persisted for decades
  - Users do not bother to install patches supplied (free) by software vendors.
- Example of vulnerability that permits remote injection of hostile code, recruiting bot nets for later DDoS attacks

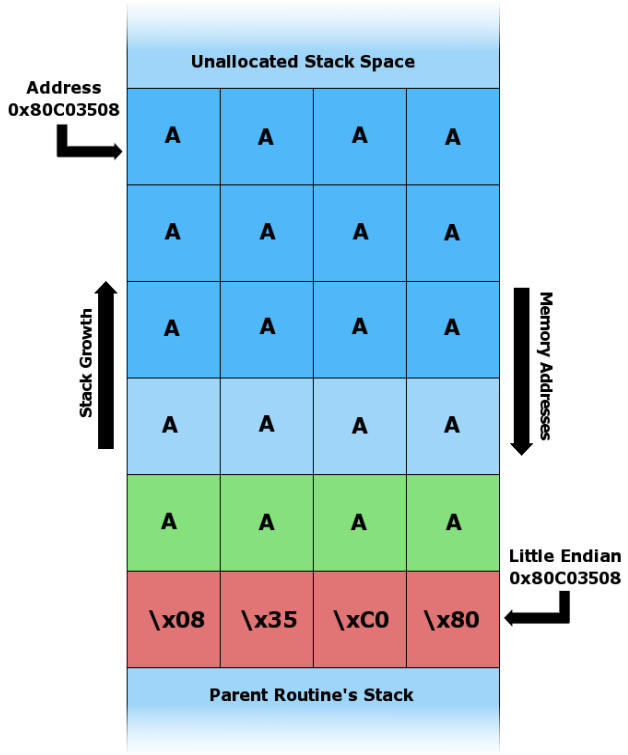
Buffer Overflow (3)



Before data is copied.



"hello" is copied.



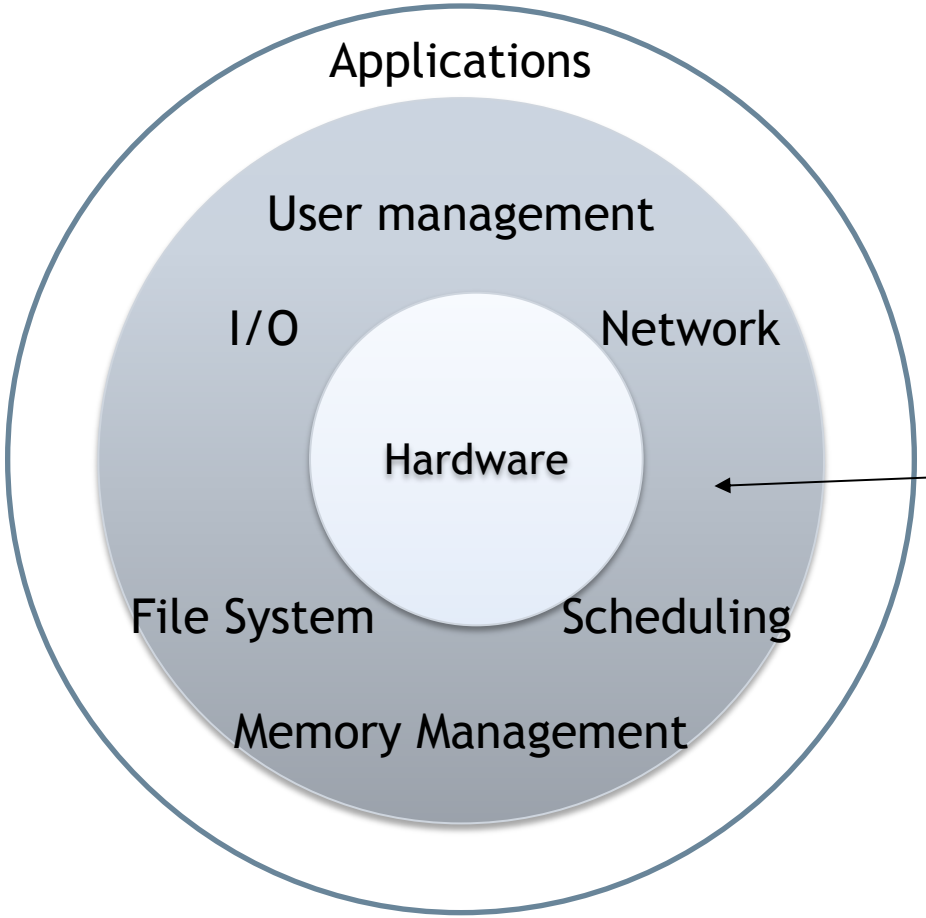
"AAAAAAAAAAAAAAAAAA\x08\x35\xC0\x80" is copied. [Wikipedia]

mobile business

Agenda

- Introduction
- Security Threats
- Operating System Security
  - Unix
  - Windows
  - Mobile OSs
- Mobile Malware
- Improving Security

OS Security Considerations



The whole code which is executed in the kernel is very security critical (supervisor mode!).

OS Kernel

It is the target of a lot of attacks to gain root privileges by manipulating kernel functions.

OS Security Control Mechanisms

- Identification
  - Recognition of human individuals
- Authentication
  - Secure confirmation of users' identifiers
- Access Control
  - Restricting usage of a service to authorized users
- Audit
  - Monitoring of system activities

mobile business

Agenda

- Introduction
- Security Threats
- Operating System Security
  - Unix
  - Windows
  - Mobile OSs
- Mobile Malware
- Improving Security

Unix Security Paradigm

- In the Unix operating system there are two parts:
  - Kernel
  - User space
- Any programming code in the kernel space has full access to the computer it is running on.
- Code running in the user space has access rights based on the User ID (UID) it is running under:
  - UID 0 is reserved for the super user or root and the kernel automatically gives this UID complete access.
- Note the difference between kernel and root access:
  - Kernel processes can access anything.
  - Root processes can order the kernel to access anything.

Unix Security Elements (1)

- IDs
  - User Identification number (UID)
  - Root/super user (UID 0)
  - Group Identification number (GID)
- Authentication
  - Password: /etc/passwd
    - User name (login name)
    - Password, encrypted
      - usually modified DES (or MD5, SHA…)
      - One way function, it is impossible to decrypt the password.
      - At login the entered password is encrypted and compared to file.
    - User id (number)
    - Login group id (number)
    - GCOS (Comment, usually real-life name)
    - Home directory
    - Program to be executed at login, usually shell

Unix Security Elements (2)

- Access Control
  - A file has owner and group id (sometimes several).
  - A process has owner and group id (sometimes several).
  - Kernel verifies permissions before executing system calls:
    - If owner uid=0 (root), everything is allowed
    - Otherwise the uid and gid of the process and object are compared in this order and permission for the operation is searched for based on owner, group and other (world) rights
- Auditing
  - Permanent Logging
  - Automatically recorded events
  - Manually set logging

mobile business

Agenda

- Introduction
- Security Threats
- Operating System Security
  - Unix
  - Windows
  - Mobile OSs
- Mobile Malware
- Improving Security

Windows Security Model

- Provides security controls access and auditing
- Implements the standard subject/object security model
  - Subject - process or thread running on behalf of the system or an authenticated user
  - Object – individually secured entity such as a file, pipe, or even a process. Access control may vary between different objects.
  - Kernel mode, User mode
- Controls applied to core OS elements like processes and sockets in addition to the more tradition file system elements (NTFS).
- Problems
  - Unexpected use of extensible elements like word macros or extensible DLL's
  - Unprotected file systems
  - Attempts at backwards compatibility with older version of Windows caused some security problems (NetBIOS and FAT).

Windows Security Elements (1)

- Identification
  - User Account
  - Security ID (SID) – A globally unique ID that refers to the subject (user or group)
- Authentication
  - Password, stored as hash value
  - Secure attention sequence CTRL+ALT+DEL
  - Security Accounts Manager
- Access Control
  - Object – Individually secured entity such as a file, pipe, or even a process
  - Rights – actions associated between object and subject (Read, write, execute, audit)
  - Access token – the runtime credentials of the subject
  - User Access Control (UAC) – administrative privileges not available by default at all times, but only after confirmation via UAC dialog box.

Windows Security Elements (2)

- Access Control
  - Access control list (ACL)
    - Associated with an object
    - Ordered list
    - Each access control entry (ACE) contains a subject and a right.
    - Evaluated by the security subsystem to determine access to protected objects
    - Discretionary ACLs control access
    - System ACLs control audit
- Auditing
  - Security Reference Monitor
  - Local Security Authority
  - Event Logger
  - If auditing applies and what is to be audited is determined by the Audit Policy

Agenda

- Introduction
- Security Threats
- Operating System Security
  - Unix
  - Windows
  - Mobile OSs
    - Mobile Linux
    - BlackBerry
    - Apple iOS
    - Android
    - Windows Phone 8
    - Firefox OS
- Mobile Malware
- Improving Security

mobile business

Once upon a time…

- Closed platforms
- No additional software could be installed.
- Limited functionality
- E.g., Java ME: restricted access

Mobile Devices Today (1)

- Open platforms
- Lots of software can be installed:
  - For different purposes
  - From different vendors
  - From different markets
- Improved hardware
  - Lot of power: quad-cores
  - Memory
- Can communication with different protocols:
  - GSM/GPRS, UMTS, 3G, 4G
  - Bluetooth, Infrared, WLAN, NFC

Mobile Devices Today (2)

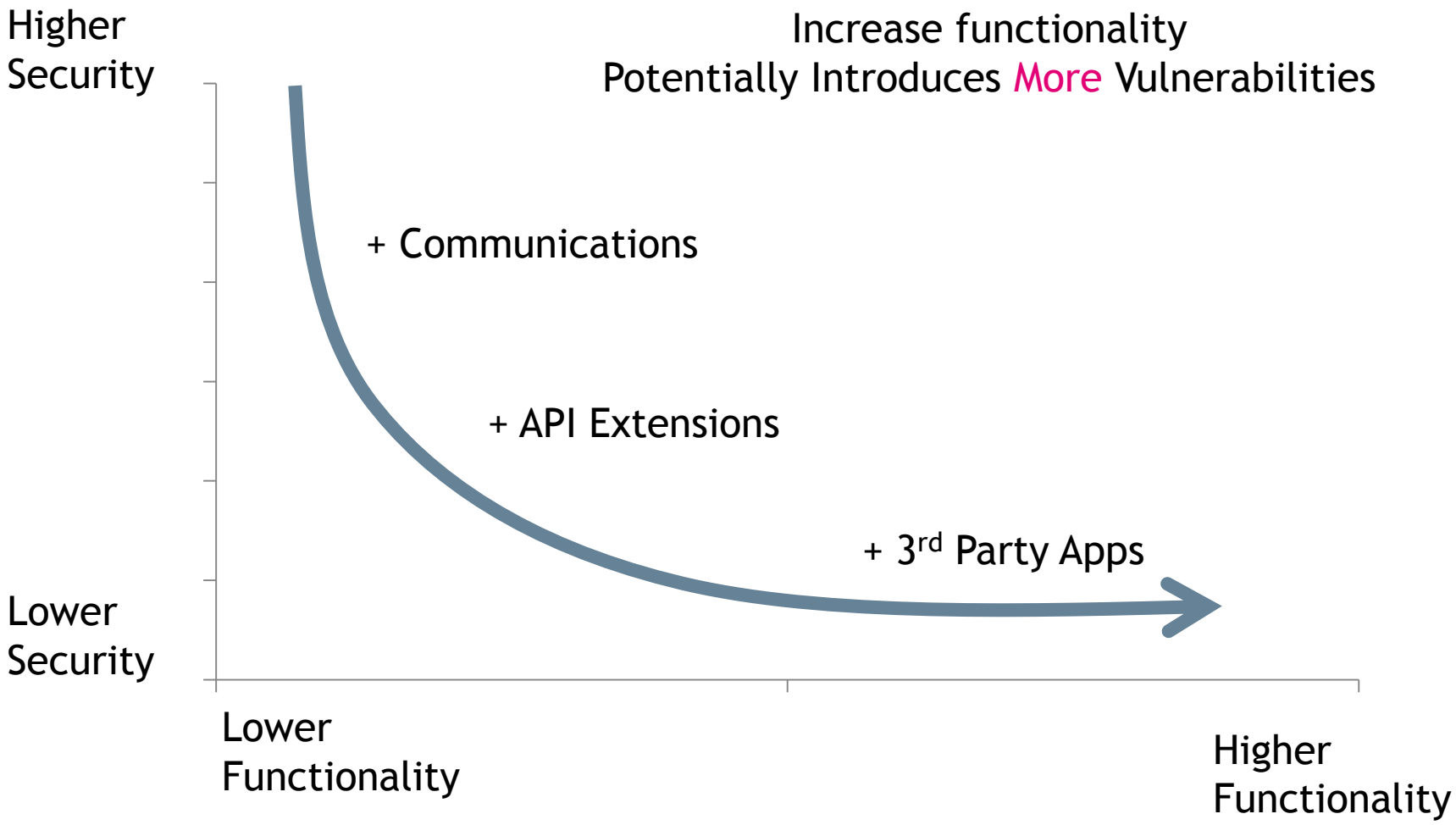
More hardware

- Camera is (in most cases) included
- Sensors
- Geolocation functionality (GPS)

Multi purpose

- Mobile: banking, commerce, …
- As authentication mechanism: OTP
- To store private and confidential data: contacts, pictures, medical records, …
- BYOD: company sensitive information

Functionality vs Security



Mobile Devices Today — Risks

- Risks of Malware
  - Viruses, Worms, Trojan Horses, etc.
- Unwanted applications
  - Adware, Trackware, Spyware
- Private data leakage
  - Location
  - Contacts
  - Device usage behavior
  - Internet surfing behavior
  - Picture, audio, video
- Always connected
  - Can perform illegal actions silently: bot

Mobile Linux

- Provides user with largest number of security functions
- Security gaps exist, while not being as prominent.
- 2005:
  - *"We think that 2005 is going to be a real breakout year for Linux on cellphones,"* Trolltech CEO Haavard Nord told *LinuxDevices.com*. *"There seems to be a huge interest in Linux. Linux is just now getting mature for the market. Currently, we're working with more than 20 manufacturers [who are] building Linux phones today,"* he said. *[LinuxDevices.com]*
- 2010:
  - Nearly every manufacturer
  - E.g. Samsung i7500
  - For details see:
  - www.linuxfordevices.com/c/a/Linux-For-Devices-Articles/Linux-Mobile-Phones/

BlackBerry

- BlackBerry became a big problem in some countries:
  - United Arab Emirates: BlackBerrys are a "national security risk”.
  - India
  - Saudi Arabia
- (Reuters) - More than a million BlackBerry users may have key services in Saudi Arabia and the UAE cut off after authorities stepped up demands on smartphone maker Research In Motion for access to encrypted messages sent over the device.
- According to an internal note from the Indian communications ministry seen by the Economic Times in India, BlackBerry has the infrastructure for solutions that would allow agencies to track messages and monitor internet traffic, but had not provided "the architect of the solution as well as the communication path for the service" [http://www.security-technologynews.com/news/indias-blackberry-security-concerns.html].
- All of this does not address the security of the (proprietary) Blackberry Phone platform, e.g. its operating system.

mobile business

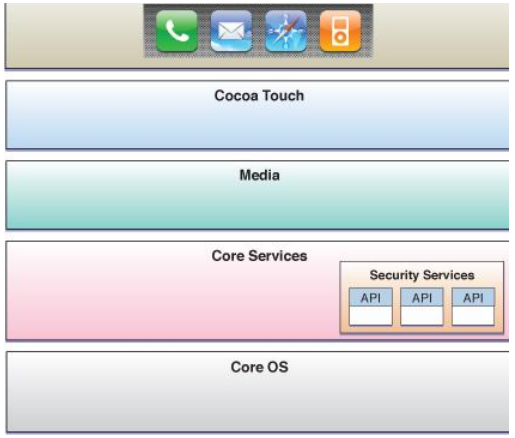
Apple iOS (1)

- The iOS security APIs are located in the Core Services layer.
- The iOS security implementation includes a daemon called the Security Server that implements several security protocols:
  - keychain items
  - root certificate trust management
- CFNetwork
  - High-level API that can be used by applications to create and maintain secure data streams and to add authentication information to a message.
- iOS provides process sandboxing:
  - An application running in iOS can see only its own keychain items and files.
- Digital signatures are required on all applications for iOS.
- Apple adds its own signature before distributing an iOS application.
- Each application is granted access permissions for certain system services when it's signed by Apple, Inc.

Source [http://developer.apple.com]

Apple iOS (2)

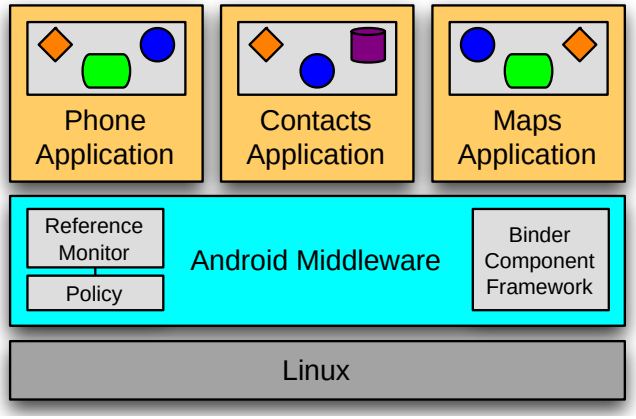
- System security
  - Secure boot chain
  - System Software authorization
- Encryption and data protection
  - Hardware security features
  - File data protection
- App security
  - App code signing
  - Runtime process security



- Network security
  - Industry standard networking protocols that provide secure authentication and encryption of data in transmission (SSL, TLS, VPN, Single Sign-on).
- Internet services
  - Apple's network-based infrastructure for messaging, syncing and backup
- Device controls
  - Methods that prevent unauthorized use of device and enable it to be remotely wiped if lost or stolen
  - Passcode protection

Android

- A Linux platform programmed with Java and enhanced with its own security mechanisms tuned for a mobile environment
- Each application declares which permission it requires at install time.
- Android *permissions* are rights given to applications to allow them to do things like:
  - directly dialling calls (which may incur tolls),
  - disclosing the user's private data, or
  - destroying address books, email, etc.



mobile business

Android

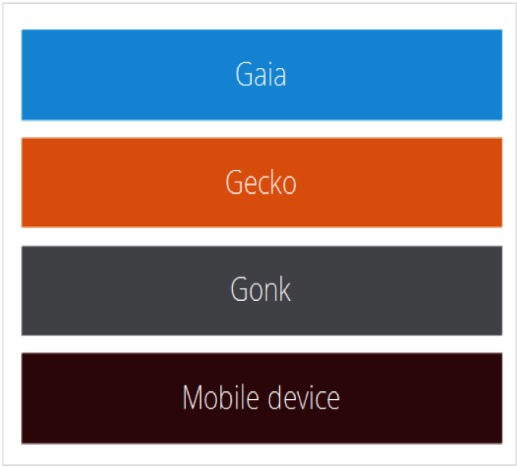
- When installed, applications are given a unique UID,
  and the application will always run as that UID on that particular device. The UID of an application is used to protect its data and developers need to be explicit about sharing data with other applications.
- Each process is running in its own address space (Dalvik virtual machine).
- The developer signs application .apk files, and the package manager verifies them.

Microsoft Windows Phone 8

- Data encryption
  - Support of several cryptographic algorithms, including AES, RSA, SHA1, SHA256,HMACSHA1, HMACSHA256, Rfc2898DeriveBytes
- Secure sockets layer (SSL) certificates
  - The Windows phone internet explorer shows a warning or error if the certificate is not valid or not issued by a trusted authority
- App management by Windows Phone app platform
  - Protect end user experience, especially
    - Avoid, that apps affect phone experience
    - Ensure, that a apps are easy to uninstall and that they uninstall completely
    - No access to the user's information without informing the user
    - No billable events without getting permission from the user
  - Application vetting
    - Apps required to go through the Windows Phone Store to be tested and digitally signed.
  - Application isolation
    - Developers use the Silverlight platform where the sandbox concept is used to provide an environment where applications have limited privileges.

Firefox OS (1)

- Firefox OS is an integrated technology stack consisting of four levels:
  - **Gaia**: the suite of web apps that makes up the user experience
  - **Gecko**: the application runtime layer that provides the framework for app execution
  - **Gonk**: the underlying Linux kernel, system libraries, firmware and device drivers that everything runs on top of
  - **The mobile device**: the mobile phone running the Firefox OS
- Security architecture
  - Multi-layered security model to mitigate exploitations risks at every level
  - Gecko as gatekeeper to enforce security policies designed to protect the mobile device from misuse



Source [www.developer.mozilla.org]

Firefox OS (2)

- Secure system deployment
  - Security measures are used throughout the technology stack.
  - File system privileges are enforced by Linux's access control lists (ACLs).
  - System apps are installed on a volume that is read-only (except during updates, when it is temporarily read-write).
- Secure system update
  - Update origin (verify the source location protocol:domain:port of the system update and manifest)
  - File integrity (SHA-256 hash check)
  - Code signature (certificate against a trusted root)
- App Security
  - Firefox OS limits and enforces the scope of resources that can be accessed or used by apps, while also supporting a wide range of apps with varying permission levels.

Source [www. developer.mozilla.org]

mobile business

Agenda

- Introduction
- Security Threats
- Operating System Security
- Mobile Malware
- Improving Security

Mobile Malware

- From traditional desktop to MOBILE
  - New mechanisms same purpose
    - Basic malware: very successful
    - Advanced: exploits, polymorphic code (=code mutates), botnets, crypto, …
  - Different goals
    - Fun
    - Fame???
    - $$$ MONEY $$$

Infection methods

- Social engineering: Phishing,
- Fake marketplace
- Repackaging an existing app and including the malware to generate a new infected version of the app.
- Creating a legitimate app, and after a period of time when it has a huge number of downloads, the attacker release an update with the malware in it.
- Executing a drive-by-download action, i.e. the app download their malicious payload after executing certain checks (e.g. is executed in a real device).
- Using malvertising to deceive users into clicking on a rogue advertisement that leads to an exploit.

mobile business

Example: Mobile Malware

- Eurograbber attack
  - 36 million Euros were stolen
  - > 30,000 victims
  - Netherlands, Spain and Germany
  - PC and mobiles targeted
  - Hijack two-factor authentication (SMS-OTP)

mobile business

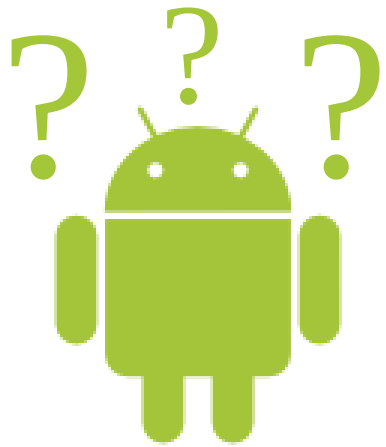
Agenda

- Introduction
- Security Threats
- Operating System Security
- Mobile Malware
- Improving Security

Security Enhancing Techniques

- **Virus scanners** try to identify viruses according to a certain characteristic (virus signature) stored in a database.
- **Code Signing** helps to distinguish authorized code from other code.
- **A Trusted Operating Base** can then prohibit the execution of not authorized code e.g. viruses on a system.
- **Checksums and/or Encryption** make it possible to detect/avoid modifications done by a virus.
- **Intrusion Detection Systems (IDS)** monitor a system to detect processes which may be the result of a virus infection.
- **Heuristic virus scanners** try to identify a virus with a forecast about the runtime behaviour of code (sophisticated approach, but not really efficient).

Questions?



Jetzabel.Serna-Olvera@m-chair.de

References

- ***J.P. Anderson**: "Computer Security Technology Planning Study,"* ESD-TR-73-51, Vols I and II, NTIS AD758206, Hanscom Field, Bedford, MA (October 1972). http://csrc.nist.gov/publications/history/ande72.pdf
- **Matt Bishop:** *Introduction to Computer Security.* Boston: Addison Wesley, 2005. pp. 363-386.
- **Dieter Gollmann:** *Computer Security.* Chichester, New York, Weinheim, Brisbane, Singapore, Toronto: John Wiley & Sons, 1999. pp. 30-54
- EMSCB -Turaya www.emscb.de/content/pages/About-Turaya-de.htm
- **National Information Systems Security (INFOSEC) Glossary**, September 2000 http://handle.dtic.mil/100.2/ADA433929
- **Intel Trusted Execution Technology** www.intel.com/technology/security
- **ISO/IEC 2382-8, Information Technology - Vocabulary - Part 8: Security, 1998**
- **LAFKON: A movie about trusted computing; 2005; www.lafkon.net/tc**
- **LinuxDevices.com** www.linuxfordevices.com
- **Microsoft's Next-Generation Secure Computing Base** www.microsoft.com/resources/ngscb/default.mspx
- iOS Security, February 2014 https://www.apple.com/au/iphone/business/docs/iOS_Security_EN_Feb14.pdf

Deutsche Telekom Chair of Mobile Business & Multilateral Security

Dr. Jetzabel M. Serna-Olvera
Goethe University Frankfurt
E-Mail: Jetzabel.Serna-Olvera@m-chair.de
WWW: www.m-chair.de