Secure Operating Systems

Nikos Tziritas

Why is OS Security an Important Issue?

- Traditional OSes focus on being easy to use without giving much attention to security
- We may open an innocent-looking email attachment and suddenly automatically run a malware on our computer
- Depending on what kind of malware it is, it might
  - show you unwanted advertisements
  - log your keystrokes
  - take over your computer
- Such an issue may jeopardize all the information stored on our computer such as:
  - Health records
  - Confidential communications

Are antivirus and firewalls enough to tackle security issues in our computers?

- Conventional security approaches like antivirus programs and firewalls are no longer enough to keep out sophisticated attackers
- It is common for malware creators to check to whether their malware is recognized by an antivirus program.
  - In such a case they modify the code such that to be non recognizable by known antivirus programs.
- Such antivirus programs will subsequently get updated once the antivirus programmers discover a new threat.
- However this will take some days to happen and new attacks may take place.
- By then it will be too late for those that have been already compromised.

Qubes OS

What is Qubes OS?

- Qubes OS is a security-oriented operating system (OS). A program managing the computer hardware
- Qubes takes an approach called security compartmentalization.
- This allows us to compartmentalize various parts of our digital life into securely isolated compartments called qubes.

Why compartmentalization?

- Such an approach will allow us to keep different things we do on our computers securely separated from each other in isolated qubes.
- In that way if one qube gets compromised will not affect the rest qubes.

A qubes example (1/2)

- We may have one qube for visiting untrusted websites and a different qube for doing online banking.
- In that way if our untrusted browsing qube gets compromised by a malware, our online activities will not be at risk.

A qubes example (2/2)

- If we are concerned about malicious email attachments, Qubes can make it so that every attachment gets opened in its own single-use disposable qube.
- In that way Qubes allows you to do everything on the same physical computer without having worrying about a single successful cyberattack taking down your entire digital life.

A Single Usable System

- All of these isolated qubes are integrated into a single usable system.
- Programs are isolated in their own separate qubes, but all windows are displayed in a single, unified desktop environment with colored window borders.
- Therefore, you can easily identify windows from different security levels.

Hardware qubes

- Common attack vectors like network cards and USB controllers are isolated in their own hardware qubes
  - Their functionality is preserved through secure networking, firewalls, and USB device management
- Integrated file and clipboard copy and paste operations make it easy to work across various qubes without compromising security.

Qubes OS vs VMs in a conventional OS

- Not all virtual machine software is equal when it comes to security.
- Type II hypervisors run under popular Oses like windows
- However, the fact that Type II hypervisors run under the host OS means that they are really only as secure as the host OS itself.
- If the host OS is ever compromised, then any VMs it hosts are also effectively compromised.

GrapheneOS

GrapheneOS

- GrapheneOS is based on the Android Open Source Project (AOSP)
- GrapheneOS makes substantial improvements to both privacy and security against other mobile operating systems
- GrapheneOS does not bundle Google apps into the OS

Attack Surface Reduction

- This is the first line of defense of GrapheneOS
- GrapheneOS removes unnecessary code or exposed attack surface to eliminate many vulnerabilities.

Preventing an attacker from exploiting vulnerabilities

- Mainstream operating systems usually don't prioritize security over other areas
- It takes an enormous amount of resources to develop fundamental fixes for these problems and there's often a high performance, memory or compatibility cost to deploying them
- GrapheneOS is willing to go further and offer toggles for users to choose the compromises they prefer instead of forcing it on them

sandboxing at various levels

- fine-grained sandboxes around a specific context like per site browser renderers
- sandboxes around a specific component like Android's media codec sandbox
- app / workspace sandboxes like the Android app sandbox used to sandbox each app which is also the basis for user/work profiles
- GrapheneOS improves all of these sandboxes through fortifying the kernel and other base OS components along with improving the sandboxing policies

Patching

- GrapheneOS includes fixes for a large number of vulnerabilities not yet fixed in Android

Sandboxed Google Play

- GrapheneOS has a compatibility layer providing the option to install and use the official releases of Google Play in the standard app sandbox
- Google Play receives absolutely no special access or privileges on GrapheneOS as opposed to bypassing the app sandbox and receiving a massive amount of highly privileged access

Google Play Apps

- Since the Google Play apps are simply regular apps on GrapheneOS, you install them within a specific user or work profile and they're only available within that profile
- Only apps within the same profile can use it and they need to explicitly choose to use it
- It works the same way as any other app and has no special capabilities. As with any other app, it can't access data of other apps and requires explicit user consent to gain access to profile data or the standard permissions

User Installed Apps can be disabled

- GrapheneOS adds support for disabling user installed apps instead of only being able to disable system apps
- This allows users to completely prevent one of the apps they've installed from being able to run without being forced to uninstall it and lose their app data

Network Permission Toggle

- GrapheneOS adds a Network permission toggle for disallowing both direct and indirect access to any of the available networks
  - Direct access is a connection to the internet through 4G/5G
  - Indirect access is a connection to the internet through a wifi router.
- The standard INTERNET permission used as the basis for the Network permission toggle is enhanced with a second layer of enforcement and proper support for granting/revoking it on a per-profile basis

Sensors permission toggle

- Sensors permission toggle: disallow access to all other sensors not covered by existing Android permissions (Camera, Microphone, Body Sensors, Activity Recognition) including an accelerometer, gyroscope, compass, barometer, thermometer and any other sensors present on a given device.
- When access is disabled, apps receive zeroed data when they check for sensor values and don't receive events. GrapheneOS creates an easy to disable notification when apps try to access sensors blocked by the permission being denied

MAC Randomization

- MAC randomization helps ensure the privacy of your mobile device by concealing the original MAC address, making it significantly harder to track a device based on its MAC address
- MAC randomization is a process that hides the exact identity of a mobile device

WI-FI Privacy

- GrapheneOS supports per-connection MAC randomization and enables it by default
- This is a more private approach than the standard persistent per-network random MAC used by modern Android.