Hands-On Ethical Hacking and Network Defense, Edition 4

Chapter 9: Embedded Operating Systems: The Hidden Threat

Module Objectives

- By the end of this module, you should be able to:
  - Explain what embedded operating systems are and where they’re used
  - Describe the Internet of Things (IoT) and other embedded operating systems
  - Identify vulnerabilities of embedded operating systems and best practices for protecting them

Introduction to Embedded Operating Systems (1 of 3)

- **Embedded system**
  - Any computer system that isn't a general-purpose PC or server
    - Found in GPS devices, ATMs, and in a wide array of electronic consumer and industrial items
- **Embedded operating system (OS)**
  - Small program developed specifically for use with embedded systems
  - Stripped-down version of OS commonly used on general-purpose computers
    - Designed to be small and efficient

Introduction to Embedded Operating Systems (2 of 3)

- **Real-time operating system (RTOS)**
  - Typically used in devices such as programmable thermostats, appliance controls, and spacecraft
- Corporate buildings
  - May have many embedded systems
    - Firewalls, switches, routers, web-filtering appliances, network attached storage (NAS) devices, etc.
- Embedded systems
  - Are in all networks
  - Perform essential functions
    - Route network traffic
    - Block suspicious packets

Introduction to Embedded Operating Systems (3 of 3)

- Recently, security researchers reverse-engineered the software on a popular firewall's chipset (**firmware**)
  - They inserted modified software to control the firewall's behavior
  - Hackers who could do this could modify a firewall to copy network traffic passing through an interface
    - Could give an external IP address full access through the firewall

Knowledge Check Activity 9-1

An embedded OS must be developed specifically for use with embedded systems. True or false?

a. True
b. False

Knowledge Check Activity 9-1: Answer

An embedded OS must be developed specifically for use with embedded systems. True or false?

Answer: b. False

An embedded operating system can be a small program developed specifically for use with embedded systems, or it can be a stripped-down version of an OS commonly used on general-purpose computers.

Polling Activity 9-1

Which of the following describes an RTOS?

a. An embedded OS capable of multitasking and responding predictably
b. An embedded OS intended for real-time data manipulation
c. An embedded OS intended for packet analysis
d. An embedded OS intended for devices that run multiple OSs

Polling Activity 9-1: Answer

Which of the following describes an RTOS?

Answer: a. An embedded OS capable of multitasking and responding predictably

RTOS is designed with an algorithm aimed at multitasking and responding predictably.

Polling Activity 9-2

Which of the following does not use an embedded OS?

a. ATM
b. Workstation running Windows 10
c. NAS device running Windows Server 2012 for Embedded Systems
d. Slot machine

Polling Activity 9-2: Answer

Which of the following does not use an embedded OS?

Answer: b. Workstation running Windows 10

A workstation running Windows 10 uses a desktop PC that does not require an embedded operating system as it is running on a Windows OS.

Windows and Other Embedded Operating Systems (1 of 2)

- Recycling common code and reusing technologies
  - Sound software engineering practices
    - However, they introduce common points of failure in many products
    - Viruses, worms, Trojans, and other attack vectors take advantage of shared code
- Windows and Linux vulnerabilities
  - Might also exist in embedded version
- Windows CE
  - Trimmed-down version of the Windows desktop OS
    - Much of the rest of it is available to hardware vendors, partners, and developers, based on their licensing level
  - Some source code is available to the public
    - Rare in 2022, but still worth knowing about

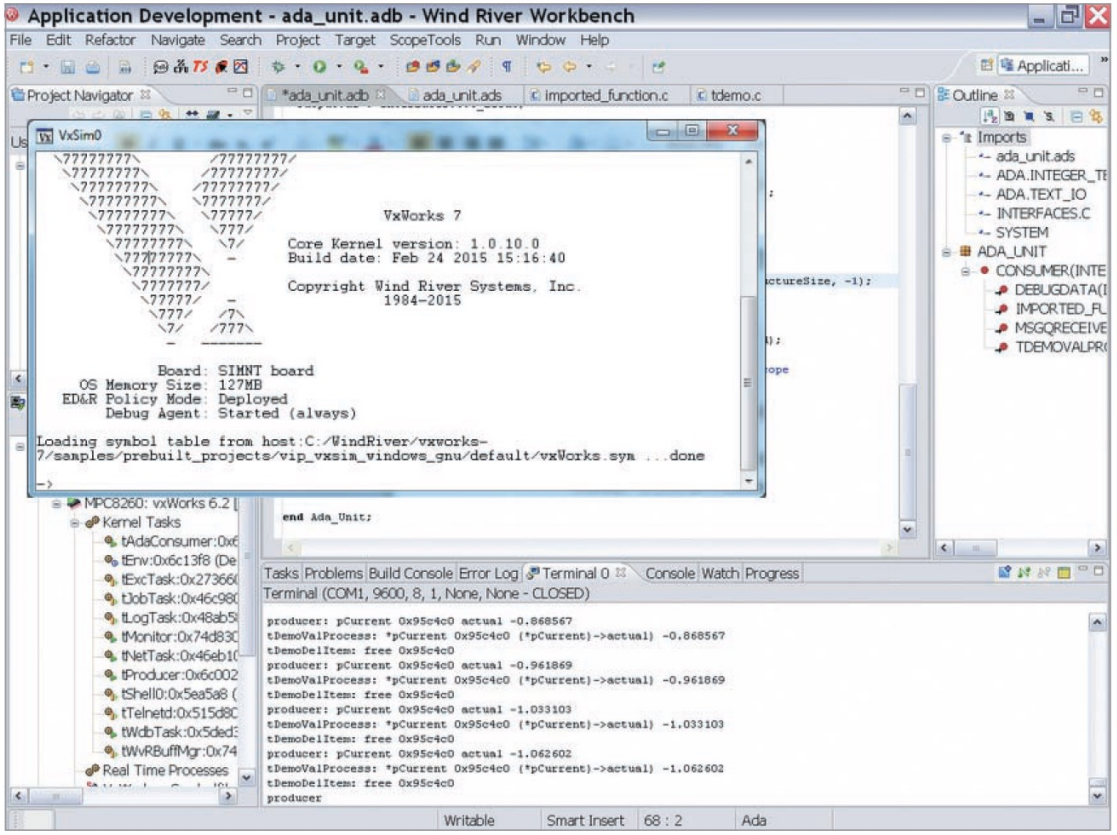
Windows and Other Embedded Operating Systems (2 of 2)

- Windows 10 IoT
  - Provides full Windows API
  - Performs many of the same tasks as the personal computer (PC) version
    - Does not have a PC interface
  - Designed for use on commodity devices
    - Example: Raspberry Pi
  - Designed to make things easy for developers

Other Proprietary Embedded OSs (1 of 5)

- VxWorks
  - Widely used embedded real-time OS
    - Developed by Wind River Systems
  - Used in many different environments and applications
  - Designed to run efficiently on minimal hardware
  - Used by a variety of systems

Other Proprietary Embedded OSs (2 of 5)



Source: Wind River Systems

Figure 9-1 Creating an embedded OS image in Wind River's VxWorks Workbench

Other Proprietary Embedded OSs (3 of 5)

- Green Hill Software produces a variety of embedded OSs
  - F-35 Joint Strike Fighter
  - **Multiple independent levels of security/safety (MILS)**
    - Embedded OS certified to run multiple levels of classification
  - Designs embedded OS code used in printers, routers, switches, etc.
- QNX Software Systems QNX
  - Commercial RTOS
  - Used in Cisco's ultrahigh-availability routers and Logitech universal remotes

Other Proprietary Embedded OSs (4 of 5)

- Real-Time Executive for Multiprocessor Systems (RTEMS)
  - An open-source embedded OS
  - Used in space systems
    - Support processors designed specifically to operate in space
- Using multiple embedded OSs increases the attack surface
  - NASA has improved Mars Reconnaissance Orbiter's survivability by using several small embedded OSs tailored for specific functions

Other Proprietary Embedded OSs (5 of 5)

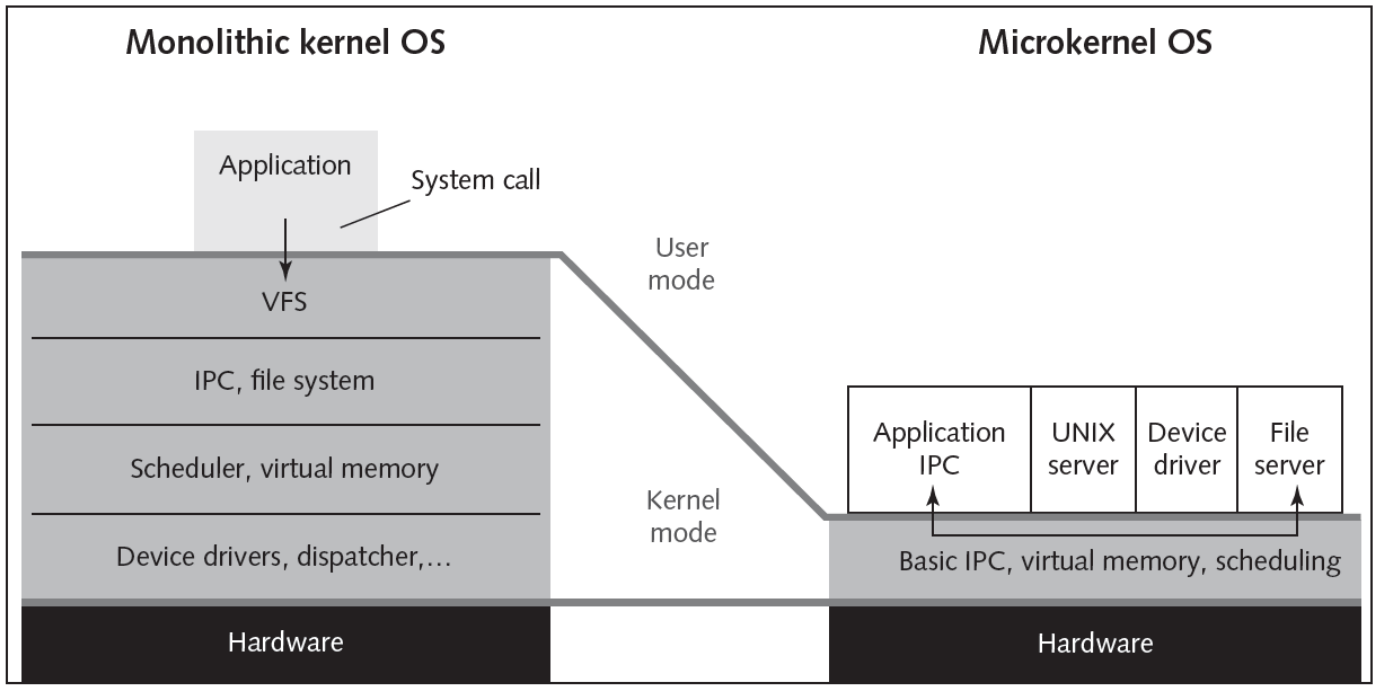


Figure 9-2 Monolithic kernel versus microkernel OSs

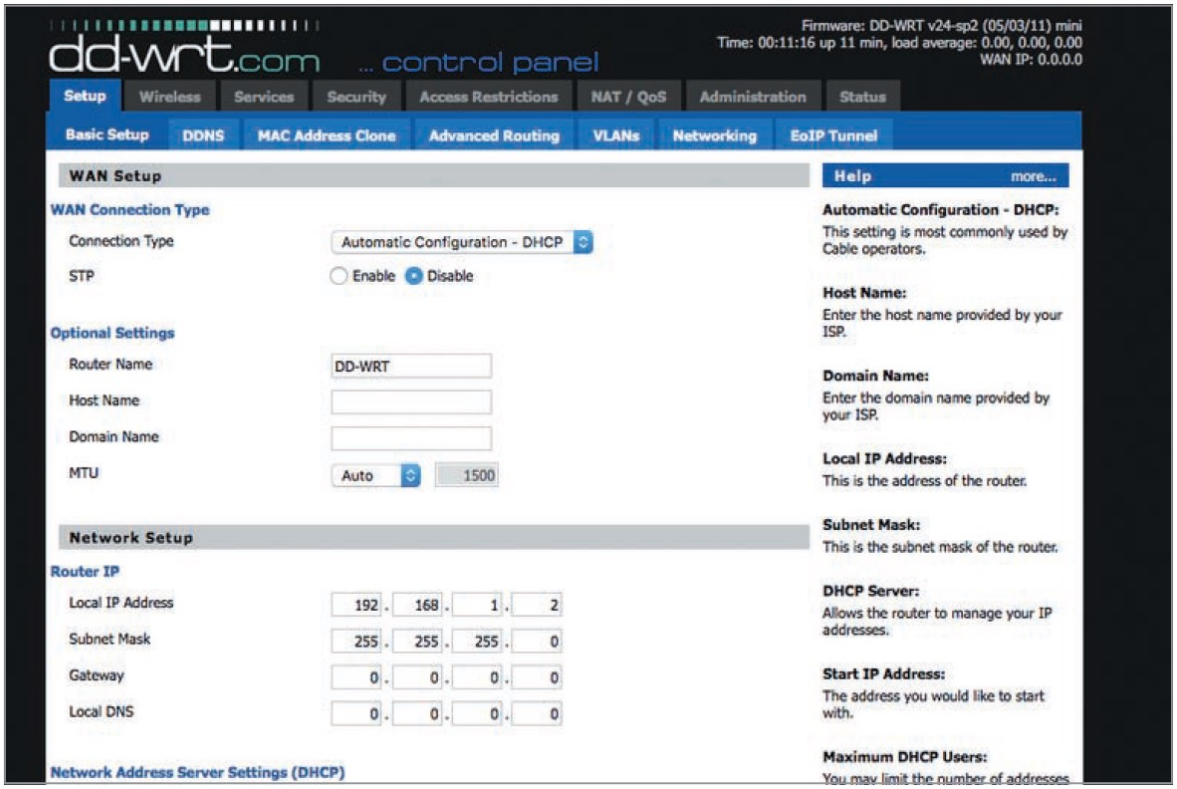
*Nix Embedded OSs (1 of 3)

- Embedded Linux
  - Example of a monolithic OS used in industrial, medical, and consumer items
  - Embedded versions of Linux and other *nix OSs can be tailored for devices with limited memory or hard drive capacity
  - Advantage of monolithic kernel
    - Supports the widest variety of hardware
    - Allows adding features by using dynamic kernel modules

*Nix Embedded OSs (2 of 3)

- Wind River Linux
  - Linux OS for embedded systems
  - Produced by Wind River
  - Linux variant RTOS suitable for embedded applications
    - Requires a guaranteed response in a mathematically predictable manner
- Linux dd-wrt
  - Embedded Linux OS
  - Initially designed for use on the Linksys WRT54G wireless router
  - Can be run on most small office or home routers

*Nix Embedded OSs (3 of 3)



Source: GNU Public License

Figure 9-3 Control panel for dd-wrt

Knowledge Check Activity 9-2

VxWorks is which of the following?

a. Windows embedded OS
b. Proprietary embedded OS
c. Linux embedded OS
d. Windows security validation tool

Knowledge Check Activity 9-2: Answer

VxWorks is which of the following?

Answer: b. Proprietary embedded OS

VxWorks is a proprietary embedded real-time OS developed by Wind River Systems.

Vulnerabilities of Embedded OSs

- Impact of attacks have become more serious
  - Embedded OSs are no exception
- Easiest way to profit from hacking
  - Attack devices that store and dispense cash (e.g., ATMs)
  - ATMs are vulnerable to attacks upon their software performed locally or remotely
    - Physical attacks can also occur, which involve the use of card skimmers or even stealing the machines

Embedded OSs Are Everywhere

- Embedded systems with Y2K software flaw were located everywhere
  - Today, there are many more embedded devices to be concerned about than in 2000
    - Are under attack from hackers and terrorists who want to further their financial or political causes
    - Addressing security early in the design phase is essential

Embedded OSs Are Networked (1 of 2)

- Advantages of connecting to a network
  - Efficiency and economy
  - Ability to manage systems and share services while keeping the amount of human resources and expertise is minimal
    - Helps companies reduce costs
- Gaining efficiency and reducing costs have a price
  - Any device added to a network infrastructure increases the potential for security problems

Embedded OSs Are Networked (2 of 2)

- Security testers should address questions such as the following:
  - What Peripheral Component Interconnect (PCI) or USB devices are present?
  - Where were they manufactured? Is the supply chain trustworthy?
  - Which devices have embedded OSs stored in rewriteable (nonvolatile) memory? Rewriteable memory can be flashed
  - Which embedded OS is currently loaded on each device?
  - Can you make sure the embedded OS hasn't been corrupted or subverted with malicious code?

Embedded OSs Are Difficult to Patch (1 of 2)

- General-purpose desktop OSs
  - Wait for vulnerability to be identified, then download, and install the patch when it's available
  - Restart the system, if necessary
  - Patching is simple
- Embedded OSs
  - Must continue operating regardless of threats, particularly in critical systems
  - Patching may be a problem
    - System administrators may have no clue how to patch a web server running on a tiny chip
  - Buffer overflow attacks might be successful
    - Because few updates are released to correct vulnerabilities
    - Manufacturers typically prefer system upgrades

Embedded OSs Are Difficult to Patch (2 of 2)

- Open-source software
  - Cost of developing and patching is shared by the entire open-source community
- Patching Linux kernel
  - Cost in programmer hours is estimated at tens of billions of dollars
- Linux kernel offers flexibility and support for sophisticated features
  - Large; has many code portions that might need to be patched
- Fixing minor vulnerabilities that are expensive to fix in embedded OSs
  - Programmer must weigh the cost of fixing the vulnerability against the importance of the information the embedded system controls

Embedded OSs Are in Networking Devices (1 of 2)

- Networking devices usually have software and hardware designed to transmit information across networks
- General-purpose computers originally performed routing and switching
  - High-speed networks now use specialized hardware and embedded OSs
- Attacks that compromise a router can give attackers the complete access to every host on the network
- Attackers follow usual methods of footprinting, scanning, and enumerating the target to compromise an entire network

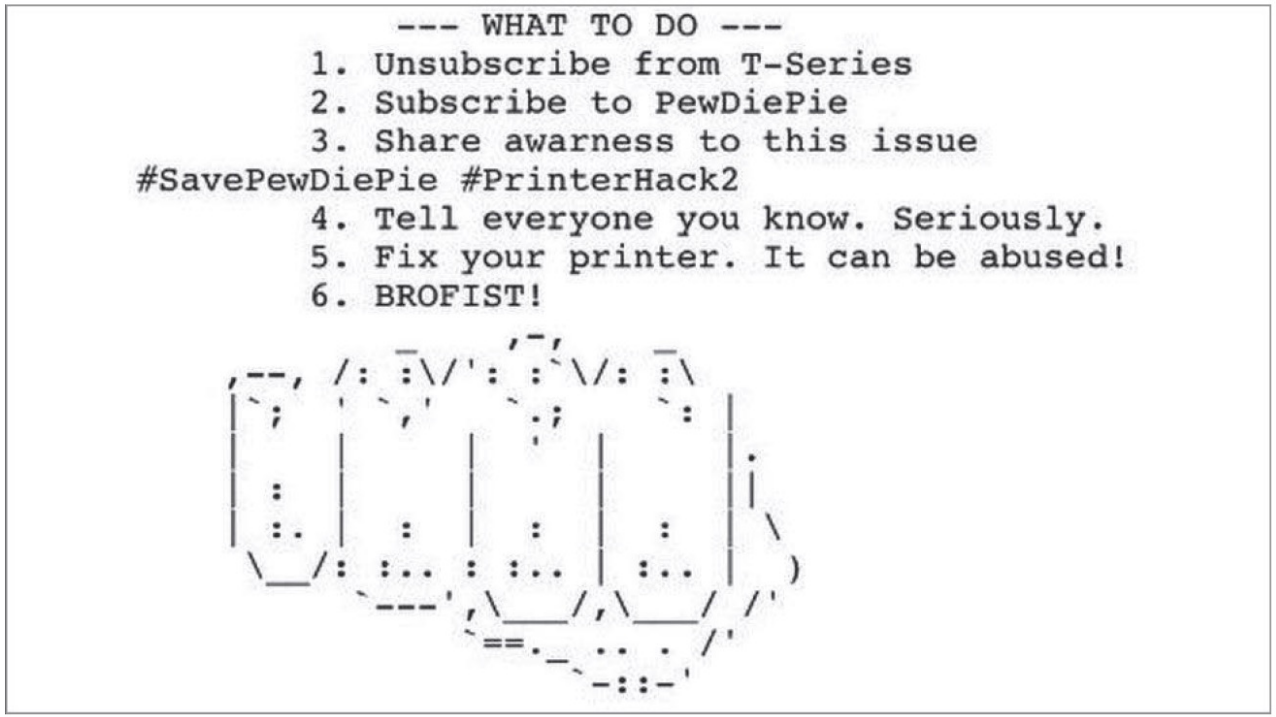
Embedded OSs Are in Networking Devices (2 of 2)

- Authentication bypass vulnerability
  - A common vulnerability of routers and other network devices with built-in web management interfaces
  - Specially crafted URL bypasses the normal authentication mechanism
- After bypassing authentication, attackers can launch other network attacks
  - They use the access gained through compromising the router

Embedded OSs Are in Network Peripherals (1 of 3)

- Common peripheral devices on an organization's network:
  - Include printers, scanners, copiers, and fax devices
- **Multifunction devices (MFDs)**
  - Perform more than one function
  - Rarely scanned for vulnerabilities or configured for security
  - Have embedded OSs with sensitive information
    - Information may be susceptible to theft and modification
    - Malware or malicious links can be inserted into the OSs
    - Social-engineering techniques may be used by attackers to masquerade as support technicians
- Printers should be reconfigured before connecting them to a network
  - Most printers have only TCP/IP enabled, but unfortunately, default administrator usernames and passwords are still configured

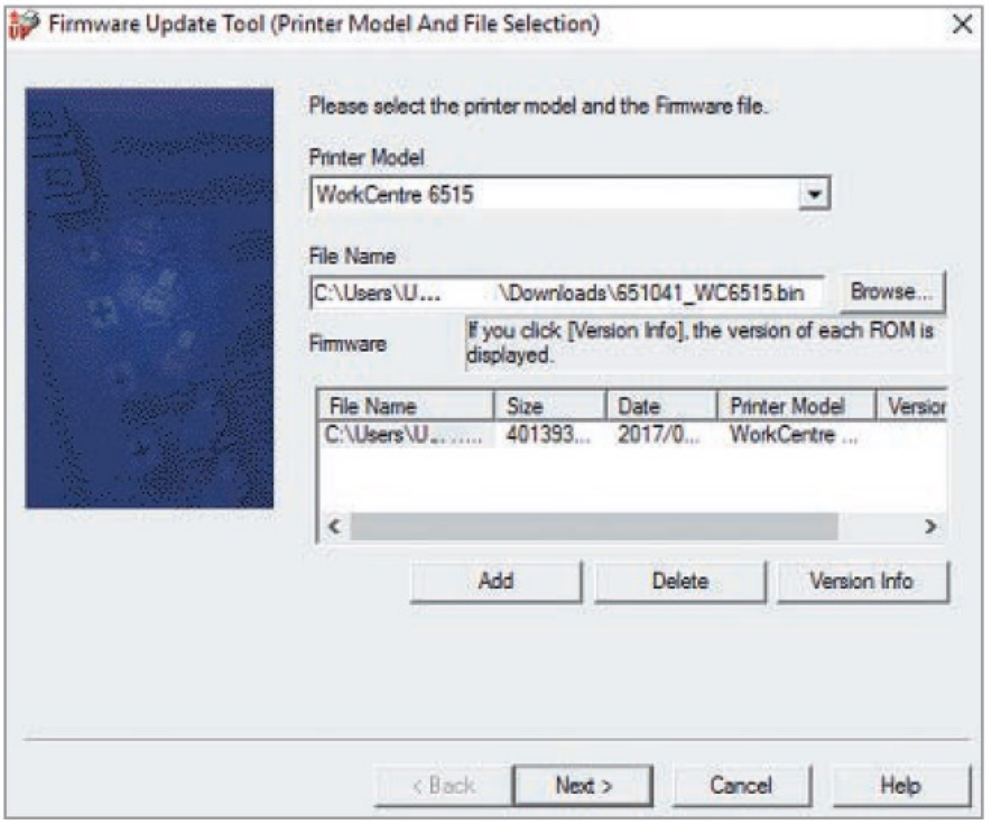
Embedded OSs Are in Network Peripherals (2 of 3)



Source: bbc.com

Figure 9-4 PewDiePie Printer Hack Output

Embedded OSs Are in Network Peripherals (3 of 3)



Source: Xerox Corporation

Figure 9-5 Firmware file being uploaded to a networked printer

Supervisory Control and Data Acquisition Systems (SCADA) (1 of 2)

- Used for equipment monitoring in large industries where automation is critical, such as public works and utilities
- May have many embedded systems as components
  - Components might be vulnerable through data fed in and out of them or through their embedded OSs
- SCADA systems controlling critical infrastructure are usually separated from the Internet by an “air gap”

Supervisory Control and Data Acquisition Systems (SCADA) (2 of 2)

- December 2015
  - An attack on a Ukrainian power plant left about 700,000 people in the dark for a few hours
  - Later, it was uncovered that a piece of malicious code named “Black-Energy” was introduced to infect systems at the power plant
    - It is assumed that attackers interacted with a SCADA system to interrupt the flow of power

Cell Phones, Smartphones, and PDAs (1 of 2)

- Conversations over traditional phones
  - Considered protected
  - Used to require a lot of time, expensive equipment, and a warrant
  - Many have the same security expectations of cell phones, smartphones, and smartphones
- Cell phone vulnerabilities include the following:
  - Attackers listening to your phone calls
  - Using the phone as a microphone
  - “Cloning” the phone to make long-distance calls
  - Get useful information for computer or network access
    - Steal trade or national security secrets

Cell Phones, Smartphones, and PDAs (2 of 2)

- Smartphones
  - Security researchers and attackers have created Java-based viruses as well as code
    - Can infect phones running Google Android, Windows Mobile, and the Apple iPhone OS (iOS)
    - The line is blurring between embedded and general-purpose OSs
      - These OSs have vulnerabilities that can compromise your smartphone's security
    - Wearable technologies increase the vulnerabilities of smartphone because of their interaction through Bluetooth
- Trojan applications have become a big concern in mobile application stores
  - Examples: Cake VPN, Pacific VPN, BeatPlayer, QR/Barcode Scanner MAX, and QRecorder

Rootkits (1 of 3)

- Modify parts of OS or install themselves as kernel modules, drivers, libraries, and applications
  - Exist for Windows and *nix OSs
- Rootkit-detection tools and antivirus software
  - Detect rootkits and prevent installation
    - More difficult if OS has already been compromised
    - Rootkits can monitor the OS for anti-rootkit tools and neutralize them
- Biggest threat
  - Rootkits that infect a device's firmware
  - More dangerous because they tend to be extremely small
    - Loaded in low-level nonvolatile storage that anti-rootkit tools can't access readily
    - Can persist even after the hard drive has been reformatted

Rootkits (2 of 3)

- Trusted Platform Module (TPM)
  - A cryptographic firmware bootcheck processor installed on many recent computer systems
  - Acts as a defense against low-level rootkits
    - Ensures that OS hasn't been subverted or corrupted
  - ISO standard ISO/IEC 11889
- Firmware rootkits
  - Hard to detect as the code for firmware often isn't checked for possible corruption
- Insider hacking
  - Harder to detect with malicious code hidden in flash memory of their company computers before they leave the company

Rootkits (3 of 3)

- Systems that are compromised even before purchase
  - May function like normal
  - Must flash (rewrite) the BIOS with a known clean copy, wipe the hard drive, and reload the OS from clean installation media
    - Expensive and time-consuming
- LoJack for Laptops
  - Popular laptop theft-recovery service
  - Has some design-level vulnerabilities that rootkits can exploit
    - Infection resides in the computer's BIOS
      - Persists even after the OS is reinstalled or the hard drive is replaced
    - Call-home mechanism
      - Allows the monitoring authority to instruct the LoJack BIOS agent to wipe all information as a security measure or to track the stolen system's location.

Best Practices for Protecting Embedded OSs (1 of 2)

- Identify all embedded systems in an organization
- Prioritize the systems or functions that depend on embedded systems
- Follow least privileges principle for access to embedded systems
- Use data transport encryption
- Configure embedded systems securely
  - Follow manufacturers' recommendations

Best Practices for Protecting Embedded OSs (2 of 2)

- Use cryptographic measures for booting embedded systems
- Install patches and updates
- Restrict network access to only the IP addresses that need to communicate with embedded systems
  - And reduce attack surface by disabling or blocking unneeded services
- Upgrade or replace embedded systems that can’t be fixed or pose unacceptable risks

Discussion Activity 9-1

Discuss as a group why rootkits that infect a device's firmware are considered the biggest threat to any OS (embedded or general-purpose).

Discussion Activity 9-1: Answer

Discuss as a group why rootkits that infect a device's firmware are considered the biggest threat to any OS (embedded or general-purpose).

Answer: Rootkits tend to be extremely small, are loaded in low-level nonvolatile storage that anti-rootkit tools can't access readily, and can persist even after the hard drive has been reformatted.

Self-Assessment

Explain why embedded device security is important.

Recall the different kinds of embedded OSs.

Summary

- Now that the lesson has ended, you should be able to:
  - Explain what embedded operating systems are and where they’re used
  - Describe the Internet of Things (IoT) and other embedded operating systems
  - Identify vulnerabilities of embedded operating systems and best practices for protecting them