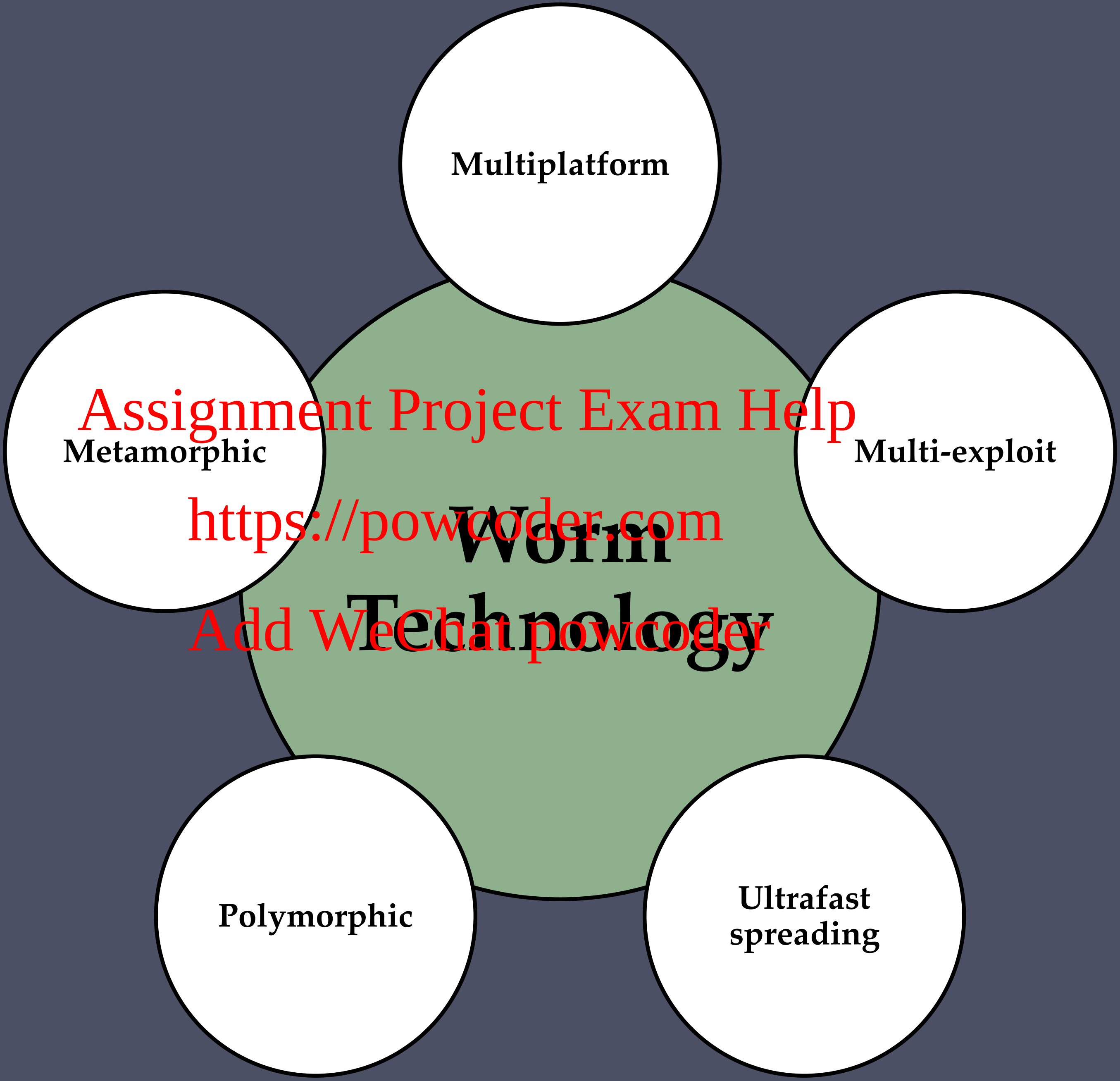
Lec – 2b
Malware: Worms

Overview

1. Worm propagation
   - ○ Worm technology
   - ○ Mobile code
   - ○ Mobile phone worms
   - ○ Drive by downloads
   - ○ Watering hole attacks
   - ○ Malvertising
   - ○ Clickjacking
2. Social engineering
3. Payloads
   - ○ System corruption
   - ○ Ransomware
   - ○ Physical damage
   - ○ Attack agents Bots
   - ○ Remote control facility
   - ○ Information theft: Keyloggers, Spywares, Phishing,
   - ○ Stealthing: Backdoor/trapdoor, rootkit
4. Countermeasures
   - ○ Approaches
   - ○ Generations of anti-virus
   - ○ Sandbox analysis
   - ○ Host-based blocking software
   - ○ Perimeter scanning approaches



Worm Technology

Multiplatform

Multi-exploit

Ultrafast spreading

Polymorphic

Metamorphic

Mobile Code

- NIST defines mobile code as
  - “programs that can be shipped unchanged to a heterogeneous collection of platforms and executed with identical semantics”
- Transmitted from a remote system to a local system and then executed on the local system
- Often acts as a mechanism for a virus, worm, or Trojan horse
- Takes advantage of vulnerabilities to perform its own exploits
- Popular vehicles include:
  - Java applets
  - ActiveX
  - JavaScript
  - VBScript
- Most common ways of using mobile code for malicious operations on local system are:
  - Cross-site scripting
  - Interactive and dynamic Web sites
  - E-mail attachments
  - Downloads from untrusted sites or of untrusted software

Mobile Phone Worms

- First discovery was Cabir worm in 2004
- Then Lasco and CommWarrior in 2005
- Communicate through Bluetooth wireless connections or MMS
- Target is the smartphone
- Can completely disable the phone, delete data on the phone, or force the device to send costly messages
- CommWarrior replicates by means of Bluetooth to other phones, sends itself as an MMS file to contacts and as an auto reply to incoming text messages

Drive-By-Downloads

Exploits browser and plugin vulnerabilities so when the user views a webpage controlled by the attacker, it contains code that exploits the bug to download and install malware on the system without the user's knowledge or consent

In most cases the malware does not actively propagate as a worm does

Spreads when users visit the malicious Web page

Watering-Hole Attacks

- A variant of drive-by-download used in highly targeted attacks
- The attacker researches their intended victims to identify websites they are likely to visit, then scans these sites to identify those with vulnerabilities that allow their compromise
- They then wait for one of their intended victims to visit one of the compromised sites
- Attack code may even be written so that it will only infect systems belonging to the target organization and take no action for other visitors to the site
- This greatly increases the likelihood of the site compromise remaining undetected

Malvertising

Places malware on websites without actually compromising them

The attacker pays for advertisements that are highly likely to be placed on their intended target websites and incorporate malware in them

Using these malicious ads, attackers can infect visitors to sites displaying them

The malware code may be dynamically generated to either reduce the chance of detection or to only infect specific systems

Has grown rapidly in recent years because they are easy to place on desired websites with few questions asked and are hard to track

Attackers can place these ads for as little as a few hours, when they expect their intended victims could be browsing the targeted websites, greatly reducing their visibility

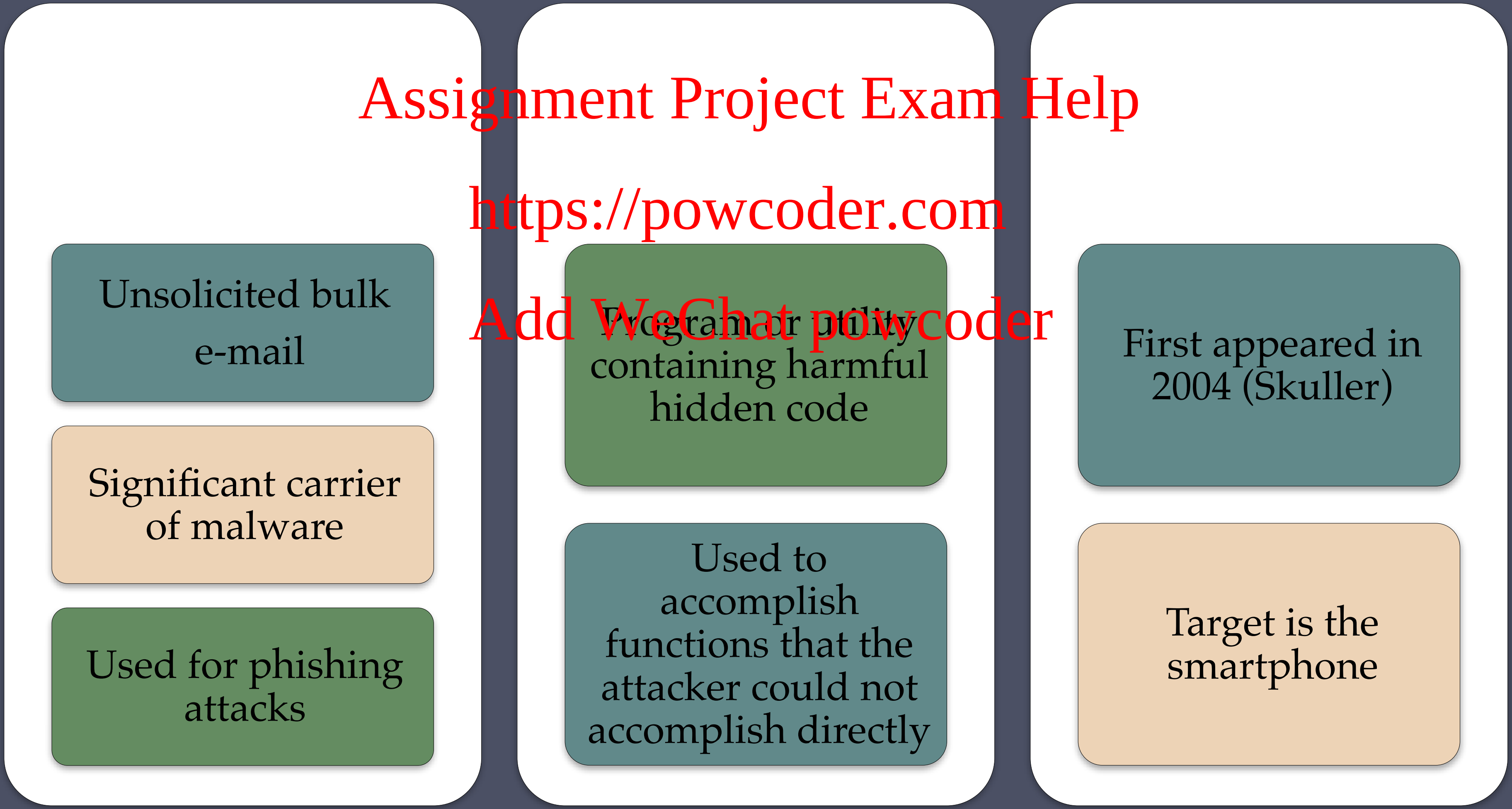
Clickjacking

- Also known as a user-interface (UI) redress attack
- Using a similar technique, keystrokes can also be hijacked
  - A user can be led to believe they are typing in the password to their email or bank account, but are instead typing into an invisible frame controlled by the attacker

- Vulnerability used by an attacker to collect an infected user's clicks
  - The attacker can force the user to do a variety of things from adjusting the user's computer settings to unwittingly sending the user to Web sites that might have malicious code
  - By taking advantage of Adobe Flash or JavaScript an attacker could even place a button under or over a legitimate button making it difficult for users to detect
  - A typical attack uses multiple transparent or opaque layers to trick a user into clicking on a button or link on another page when they were intending to click on the top level page
  - The attacker is hijacking clicks meant for one page and routing them to another page

Social Engineering

- "Tricking" users to assist in the compromise of their own systems



Unsolicited bulk e-mail

Significant carrier of malware

Used for phishing attacks

Program or utility containing harmful hidden code

Used to accomplish functions that the attacker could not accomplish directly

First appeared in 2004 (Skuller)

Target is the smartphone

Payload
System Corruption

Chernobyl virus

- First seen in 1998
- Example of a destructive parasitic memory-resident Windows 95 and 98 virus
- Infects executable files when they are opened and when a trigger date is reached, the virus deletes data on the infected system by overwriting the first megabyte of the hard drive with zeroes, resulting in massive corruption of the entire file system

Klez

- Mass mailing worm infecting Windows 95 to XP systems
- First seen in October 2001
- Spreads by emailing copies of itself to addresses found in the address book and in files on the system
- It can stop and delete some anti-virus programs running on the system
- On trigger date causes files on the hard drive to become empty

Ransomware

- Mid-2006 a number of worms and Trojans appeared that used public-key cryptography with incresasingly larger key sizes to encrypt data
- The user needed to pay a ransom, or to make a purchase from certain sites, in order to receive the key to decrypt this data
- Ransom money in Bitcoins

- WannaCry
  - Infected a large number of systems in many countries in May 2017
  - When installed on infected systems, it encrypted a large number of files and then demanded a ransom payment in Bitcoins to recover them
  - Recovery of this information was generally only possible if the organization had good backups and an appropriate incident response and disaster recovery plan
  - Targets widened beyond personal computer systems to include mobile devices and Linux servers
  - Tactics such as threatening to publish sensitive personal information, or to permanently destroy the encryption key after a short period of time, are sometimes used to increase the pressure on the victim to pay up

Payload – Attack Agents Bots

- Takes over another Internet attached computer and uses that computer to launch or manage attacks
- *Botnet* - collection of bots capable of acting in a coordinated manner
- Uses:
  - Distributed denial-of-service (DDoS) attacks
  - Spamming
  - Sniffing traffic
  - Keylogging
  - Spreading new malware
  - Installing advertisement add-ons and browser helper objects (BHOs)
  - Attacking IRC chat networks
  - Manipulating online polls/games

Remote Control Facility

- Distinguishes a bot from a worm
    - Worm propagates itself and activates itself
    - Bot is initially controlled from some central facility
- Typical means of implementing the remote control facility is on an IRC server (Internet Relay Chat (**IRC**) is an application layer protocol that facilitates communication in the form of text)
    - Bots join a specific channel on this server and treat incoming messages as commands
    - More recent botnets use covert communication channels via protocols such as HTTP
    - Distributed control mechanisms use peer-to-peer protocols to avoid a single point of failure

Payload – Information Theft Keyloggers and Spyware

Keylogge r

- Captures keystrokes to allow attacker to monitor sensitive information
- Typically uses some form of filtering mechanism that only returns information close to keywords ("login", "password")

Spyware

- Subverts the compromised machine to allow monitoring of a wide range of activity on the system
  - Monitoring history and content of browsing activity
  - Redirecting certain Web page requests to fake sites
  - Dynamically modifying data exchanged between the browser and certain Web sites of interest

Payload – Information Theft Phishing

- Exploits social engineering to leverage the user's trust by masquerading as communication from a trusted source
  - Include a URL in a spam e-mail that links to a fake Web site that mimics the login page of a banking, gaming, or similar site
  - Suggests that urgent action is required by the user to authenticate their account
  - Attacker exploits the account using the captured credentials

- Spear-phishing
  - Recipients are carefully researched by the attacker
  - E-mail is crafted to specifically suit its recipient, often quoting a range of information to convince them of its authenticity

Payload – Stealthing Backdoor

- Also known as a *trapdoor*
- Secret entry point into a program allowing the attacker to gain access and bypass the security access procedures
- *Maintenance hook* is a backdoor used by Programmers to debug and test programs
- Difficult to implement operating system controls for backdoors in applications

Payload - Stealthing Rootkit

- Set of hidden programs installed on a system to maintain covert access to that system

- Hides by subverting the mechanisms that monitor and report on the processes, files, and registries on a computer

- Gives administrator (or root) privileges to attacker
  - Can add or change programs and files, monitor processes, send and receive network traffic, and get backdoor access on demand

Rootkit Classification Characteristics

Persistent

Memory based

User mode

Kernel mode

Virtual machine based

External mode

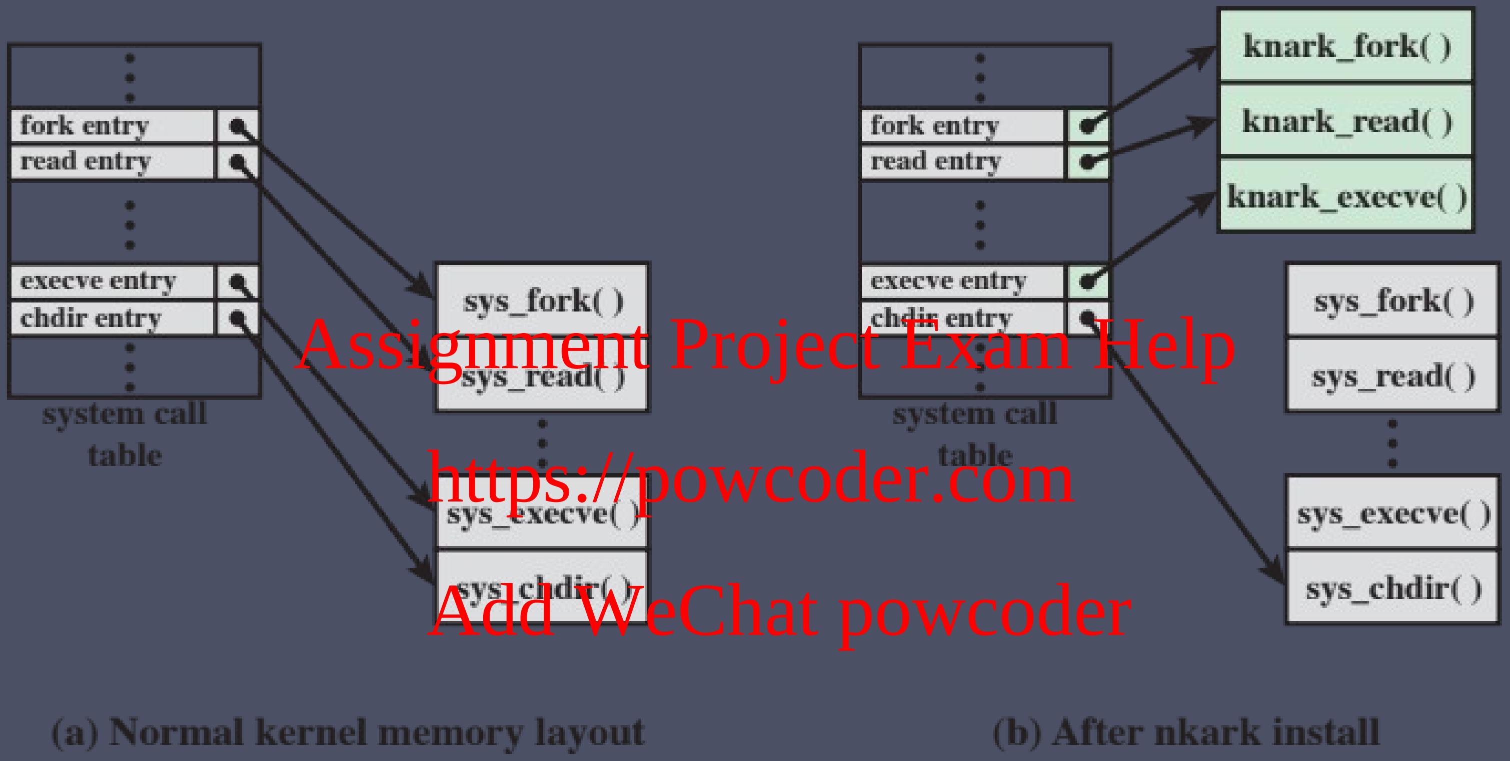


Figure 6.3 System Call Table Modification by Rootkit

Malware Countermeasure Approaches

- Ideal solution to the threat of malware is prevention

Four main elements of prevention:

- Policy
- Awareness
- Vulnerability mitigation
- Threat mitigation

- If prevention fails, technical mechanisms can be used to support the following threat mitigation options:
  - Detection
  - Identification
  - Removal

Requirements for effective malware countermeasures:

Generality: to handle a wide variety of attacks.

Timeliness: respond quickly so as to limit the number of infected programs or systems and the consequent activity.

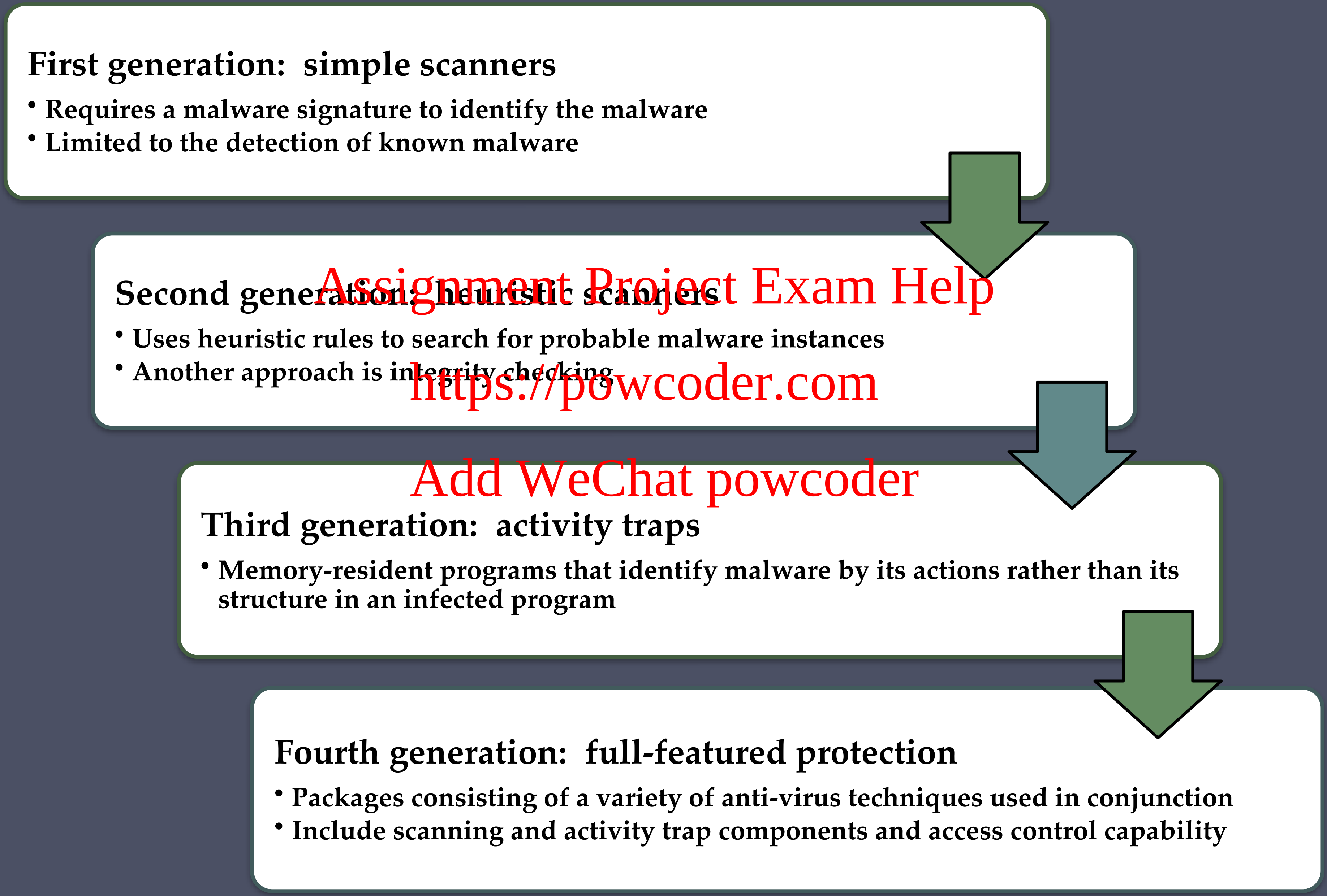
Resiliency: resistant to evasion techniques employed by attackers to hide the presence of their malware.

Minimal denial-of-service costs: minimal reduction in capacity or service due to the actions of the countermeasure software and should not significantly disrupt normal operation.

Transparency: not require modification to existing (legacy) OSs, application software, and hardware.

Global and local coverage: deal with attack sources both from outside and inside the enterprise network.

Generations of Anti-Virus Software



First generation: simple scanners

- Requires a malware signature to identify the malware
- Limited to the detection of known malware

Second generation: heuristic scanners

- Uses heuristic rules to search for probable malware instances
- Another approach is integrity checking

Third generation: activity traps

- Memory-resident programs that identify malware by its actions rather than its structure in an infected program

Fourth generation: full-featured protection

- Packages consisting of a variety of anti-virus techniques used in conjunction
- Include scanning and activity trap components and access control capability

Sandbox Analysis

- Running potentially malicious code in an emulated sandbox or on a virtual machine
- Allows the code to execute in a controlled environment where its behavior can be closely monitored without threatening the security of a real system
- Running potentially malicious software in such environments enables the detection of complex encrypted, polymorphic, or metamorphic malware
- The most difficult design issue with sandbox analysis is to determine how long to run each interpretation

Host-Based Behavior-Blocking Software

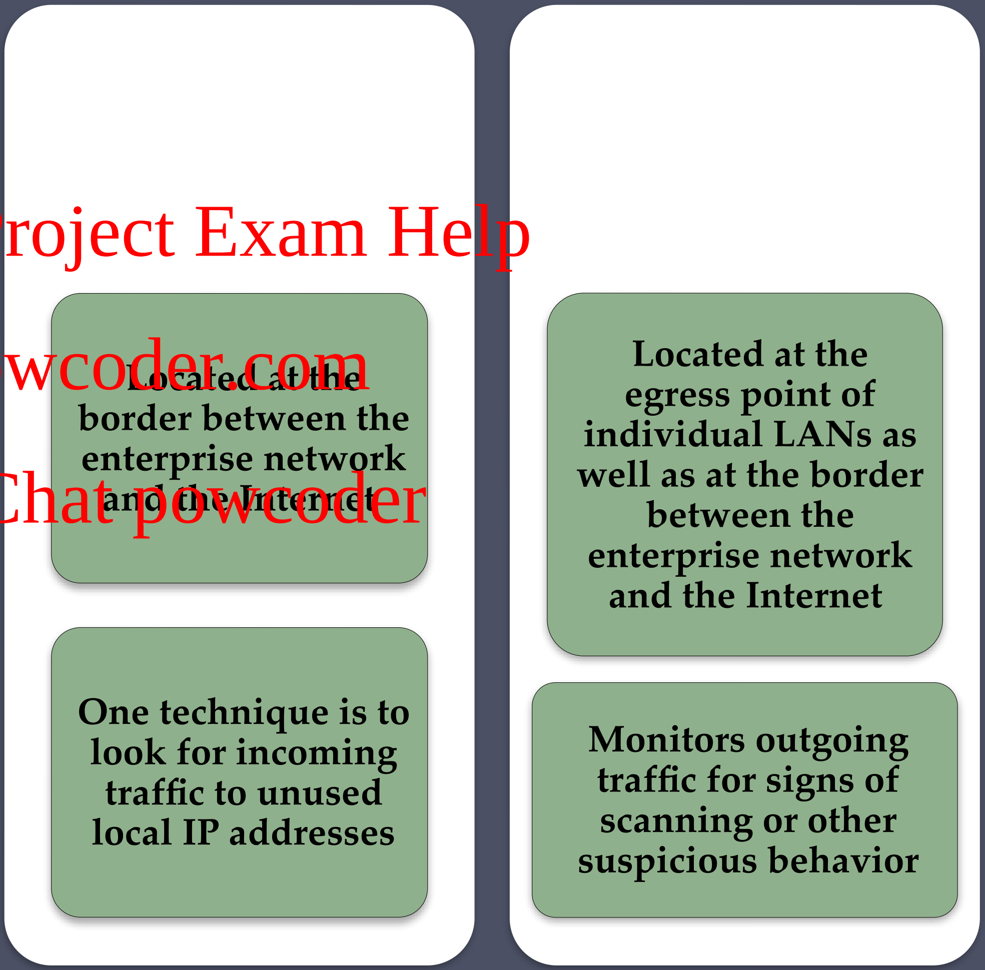
- Integrates with the operating system of a host computer and monitors program behavior in real time for malicious action
  - Blocks potentially malicious actions before they have a chance to affect the system
  - Blocks software in real time so it has an advantage over anti-virus detection techniques such as fingerprinting or heuristics

Limitations

- Because malicious code must run on the target machine before all its behaviors can be identified, it can cause harm before it has been detected and blocked

Perimeter Scanning Approaches

- Anti-virus software typically included in e-mail and Web proxy services running on an organization’s firewall and IDS
- May also be included in the traffic analysis component of an IDS
- May include intrusion prevention measures, blocking the flow of any suspicious traffic
- Approach is limited to scanning malware



Located at the border between the enterprise network and the Internet

One technique is to look for incoming traffic to unused local IP addresses

Located at the egress point of individual LANs as well as at the border between the enterprise network and the Internet

Monitors outgoing traffic for signs of scanning or other suspicious behavior

Two types of monitoring software