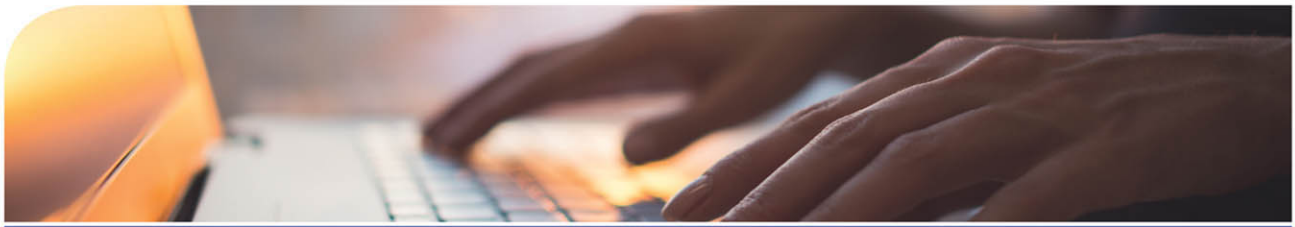
DAVID L. PROWSE

CompTIA Approved Quality Content

Cert Guide

Learn, prepare, and practice for exam success



CompTIA® Security+ SY0-501

CompTIA Security+

PEARSON IT CERTIFICATION

FEATURES

Three Complete Practice Exams, More Than 30 Videos and 30 Interactive Exercises

But programmers use port zero as a wildcard port, designing their applications to ask the operating system to assign a non-zero port. So, normally malware that exploits port zero will simply be redirected to another valid port. Again, this means that only 65,535 of the 65,536 ports can be exploited. In the future, port zero may become more of a security concern with the growth of legitimate raw socket programming. This is programming directly to network ports, bypassing the transport layer; an example would be the Internet Control Message Protocol (ICMP) involving ping operations. However, historically, raw sockets have been used by attackers to perform *TCP reset attacks*, which set the reset flag in a TCP header to 1, telling the respective computer to kill the TCP session immediately. Until recently, raw socket programming has been generally frowned upon.

Protocols That Can Cause Anxiety on the Exam

Unfortunately, a lot of the protocols look similar, behave similarly, and can be downright confusing. Let's discuss a few of the more difficult ones and try to dispel some of the confusion. We start with FTP and its derivatives.

You know about the FTP protocol and what it does. You probably also know that FTP can be inherently insecure. There are several ways to make FTP sessions more secure. We mentioned previously that you can use FTP software that randomizes which ports are selected to transfer each file. You can also select passive mode instead of active mode (most FTP clients default to passive). The difference is that in passive mode the server is required to open ports for incoming traffic, and in active mode both the server and the client open ports. Then, you could use an FTP protocol that is secured through encryption. Two examples are Secure FTP (SFTP) and FTP Secure (FTPS). SFTP uses SSH port 22 to make connections to other systems. Because of this it is also known as SSH FTP. However, FTPS works with SSL or TLS, and (in implicit mode) it uses ports 990 (control port) and 989 (data port) to make secure connections and send data, respectively. FTPS can work in two modes: explicit mode and the previously mentioned implicit mode. In explicit mode, the FTPS client must explicitly request security from an FTPS server and then mutually agree on the type of encryption to be used. In implicit mode, there is no negotiation, and the client is expected to already know the type of encryption used by the server. In general, implicit mode is considered to be more secure than explicit mode.

So, in summary, regular FTP uses port 21 as the control port by default, and possibly port 20 to do data transfers—or (and more likely), it uses random ports for data transfers, if the software allows it. SFTP uses port 22. FTPS uses port 990 to make connections, and port 989 to transfer data by default. TFTP (which is not really secure) uses port 69.

On a separate note, another file transfer program, Secure Copy (SCP), is an example of a protocol that uses an additional protocol (and its corresponding port) for security. It uses SSH, and ultimately uses port 22 to transfer data.

All those acronyms can be difficult to keep straight at times. Hopefully this section alleviates some of the confusion. For more help, be sure to memorize Table 7-2 to the best of your ability for the exam, and don't be afraid to ask me questions on my website!

Malicious Attacks

There are many types of malicious network attacks. We've mentioned some of these attacks in the preceding chapters as they relate to secure computing, but in this section we will better define them. Some attacks are similar to others, making it difficult to differentiate between them. Because of this, I've listed simple definitions and examples of each, plus mitigating techniques, and summarized them at the end of this section.

DoS

Denial-of-service (DoS) is a broad term given to many different types of network attacks that attempt to make computer resources unavailable. Generally, this is done to servers but could also be perpetuated against routers and other hosts. DoS attacks can be implemented in several ways, as listed here:

- **Flood attack:** An attacker sends many packets to a single server or other host in an attempt to disable it. There are a few ways to accomplish this, including:
  - **Ping flood:** Also known as an ICMP flood attack, this is when an attacker attempts to send many ICMP echo request packets (pings) to a host in an attempt to use up all available bandwidth. This works only if the attacker has more bandwidth available than the target. To deter this attack, configure the system not to respond to ICMP echoes. You might have noticed that several years ago, you could ping large companies' websites and get replies. But after ping floods became prevalent, a lot of these companies disabled ICMP echo replies. For example, try opening the command prompt and typing `ping microsoft.com` (Internet connection required). It should result in Request Timed Out, which tells you that Microsoft has disabled this.
  - **Smurf attack:** Also sends large amounts of ICMP echoes, but this particular attack goes a bit further. The attacking computer broadcasts the ICMP echo requests to every computer on its network or subnetwork. In addition, in the header of the ICMP echo requests will be a spoofed IP

address. That IP address is the target of the Smurf attack. Every computer that replies to the ICMP echo requests will do so to the spoofed IP. Don't forget that the original attack was broadcast, so, the more systems on the network (or subnetwork), the more echo replies that are sent to the target computer. There are several defenses for this attack, including configuring hosts not to respond to pings or ICMP echoes, configuring routers not to forward packets directed to broadcast addresses, implementing subnetting with smaller subnetworks, and employing network ingress filtering in an attempt to drop packets that contain forged or spoofed IP addresses (especially addresses on other networks). These defenses have enabled most network administrators to make their networks immune to Smurf and other ICMP-based attacks. The attack can be automated and modified using the exploit code known as Smurf.c.

— **Fraggle:** Similar to the Smurf attack, but the traffic sent is UDP echoes. The traffic is directed to port 7 (Echo) and port 19 (CHARGEN). To protect against this attack, again, configure routers not to forward packets directed to broadcast addresses, employ network filtering, and disable ports 7 and 19. These ports are not normally used in most networks. The attack can be automated and modified using the exploit code known as Fraggle.c.

NOTE A similar attack is known as a UDP flood attack, which also uses the connectionless User Datagram Protocol. It is enticing to attackers because it does not require a synchronization process.

— **SYN flood:** Also known as a SYN attack, it occurs when an attacker sends a large amount of SYN request packets to a server in an attempt to deny service. Remember that in the TCP three-way handshake, a synchronization (SYN) packet is sent from the client to the server, then a SYN/ACK packet is sent from the server to the client, and finally, an acknowledgment (ACK) packet is sent from the client to the server. Attackers attempting a SYN flood either simply skip sending the ACK or spoof the source IP address in the original SYN. Either way, the server will never receive the final ACK packet. This ends up being a half-open connection. By doing this multiple times, an attacker seeks to use up all connection-oriented resources so that no real connections can be made. Some ways to defend against this include implementing **flood guards** (which can be implemented on some firewalls and other devices, otherwise known as attack guards), recycling half-open connections after a predetermined amount of time, and using intrusion detection systems

(IDSs) to detect the attack. You can find more information about IDSs in Chapter 8 and more information about SYN flood attacks and mitigation techniques at the following link: https://tools.ietf.org/html/rfc4987.

— **Xmas attack:** Also known as the Christmas Tree attack or TCP Xmas Scan attack, it can deny service to routers and other devices, or simply cause them to reboot. It is based on the Christmas Tree packet, which can be generated by a variety of programs; for example, Nmap can be used (with the `-sX` parameter) to produce this scanning packet. This type of packet has the FIN, PSH, and URG flags set, which gives a "Christmas Tree" appearance when viewing the flags in a network sniffer. If the packet is sent many times in a short period of time, it could possibly result in a DoS (which is why I placed this attack in the DoS flood section). But most routers and other devices today will block this type of packet, as it is a well-known attack. Otherwise, an IDS/IPS solution (if in place) can detect the packet and/or prevent the packet from denying service to a router or other device.

- **Ping of Death:** POD is an attack that sends an oversized and malformed packet to another computer. It is an older attack; most computer operating systems today will not be affected by it, and most firewalls will block it before it enters a network. It entails sending a packet that is larger than 65,535 bytes in length, which according to RFC 791 is the largest size packet that can be used on a TCP/IP network without fragmentation. It should be noted that, normally, the maximum transmission unit (MTU) size of an Ethernet frame is 1500 bytes, and slightly less for the encapsulated TCP/IP packet. Going beyond this requires special means. Now, if a packet is sent that is larger than 65,535 bytes, it might overflow the target system's memory buffers, which can cause several types of problems, including system crashes. Windows computers do not allow ping sizes beyond 65,500 bytes. For example, `ping destination -l 65500` will work, but `ping destination -l 66000` will not work. However, on some systems, this maximum limitation can be hacked in the Registry, and there are also third-party applications that can send these "larger than life" packets. To protect against this type of attack, configure hosts not to respond to pings or ICMP echoes, make sure that operating systems run the latest service packs and updates, update the firmware on any hardware-based firewalls, and update any software-based firewalls as well. POD can be combined with a ping flood, but because most firewalls will block one or more PODs, it doesn't make much sense to attempt the attack, so most attackers opt for some other sort of packet flooding nowadays. This was one of the first DoS attacks. It and other attacks such as Nuke and WinNuke are considered by the security community to be deprecated.

- **Teardrop attack:** Sends mangled IP fragments with overlapping and oversized payloads to the target machine. This can crash and reboot various operating systems due to a bug in their TCP/IP fragmentation reassembly code. For example, some older versions of Windows are particularly susceptible to teardrop attacks. Linux and Windows systems should be upgraded to protect from this attack. There are also software downloads available on the Internet for teardrop detection.
- **Permanent DoS attack:** Generally consists of an attacker exploiting security flaws in routers and other networking hardware by flashing the firmware of the device and replacing it with a modified image. This is also known as phlashing, or PDoS.
- **Fork bomb:** Works by quickly creating a large number of processes to saturate the available processing space in the computer's operating system. Running processes can be "forked" to create other running processes, and so on. They are not considered viruses or worms but are known as "rabbit malware," "wabbits," or "bacteria" because they might self-replicate but do not infect programs or use the network to spread. They are still considered DoS attacks though, due to their ability to stop a system from functioning.

There are other types of DoS attacks, but that should suffice for now. Keep in mind that new DoS attacks are always being dreamed up (and implemented), so as a security administrator, you need to be ready for new attacks and prepared to exercise new mitigation techniques.

DDoS

A **distributed denial-of-service (DDoS)** attack is when a group of compromised systems attacks a single target, causing a DoS to occur at that host. A DDoS attack often utilizes a botnet—which is a large group of computers known as robots or simply "bots." Often, these are systems owned by unsuspecting users. The computers in the botnet that act as attackers are known as zombies. An attacker starts the DDoS attack by exploiting a single vulnerability in a computer system and making that computer the zombie master, or DDoS master. The master system communicates with the other systems in the botnet. The attacker often loads malicious software on many computers (zombies). The attacker can launch a flood of attacks by all zombies in the botnet with a single command. DDoS attacks and botnets are often associated with exploit kits (such as the Blackhole kit) and ransomware.

DoS and DDoS attacks are difficult to defend against. Other than the methods mentioned previously in the DoS section, these attacks can be prevented to some extent by updated stateful firewalls, switches, and routers with access control lists, intrusion prevention systems (IPSs), and proactive testing. Several companies offer products

that simulate DoS and DDoS attacks. By creating a test server and assessing its vulnerabilities with simulated DoS tests, you can find holes in the security of your server before you take it live. A quick web search for "DoS testing" shows a few of these simulation test companies. An organization could also opt for a "clean pipe," which attempts to weed out DDoS attacks, among other attacks. This solution is offered as a service by Verisign and other companies. Manual protection of servers can be a difficult task; to implement proper DDoS mitigation, your organization might want to consider anti-DDoS technology and emergency response from an outside source or from the organization's cloud-based provider. Finally, if you do realize that a DDoS attack is being carried out on your network, call your ISP and request that this traffic be redirected.

One specific type of DDoS is the **DNS amplification attack**. Amplification attacks generate a high volume of packets ultimately intended to flood a target website. In the case of a DNS amplification attack, the attacker initiates DNS requests with a spoofed source IP address. The attacker relies on *reflection*; responses are not sent back to the attacker, but are instead sent "back" to the victim server. Because the DNS response is larger than the DNS request (usually), it *amplifies* the amount of data being passed to the victim. An attacker can use a small number of systems with little bandwidth to create a sizable attack. However, a DNS amplification attack can also be accomplished with the aid of a botnet, which has proven to be devastating to sections of the Internet during the period when the attack was carried out.

The primary way of preventing this attack is to block spoofed source packets. It can also be prevented by blocking specific DNS servers, blocking open recursive relay servers, rate limiting, and updating one's own DNS server(s) often. Finally, make use of the Domain Name System Security Extensions (DNSSEC), which are specifications that provide for origin authentication and data integrity.

NOTE Smurf and Fraggle are also examples of amplification attacks.

Sinkholes and Blackholes

To combat DoS and DDoS attacks, security admins have the option to employ or make use of sinkholes, blackholes, and blackhole lists. A DNS sinkhole is a DNS server that can be configured to hand out false information to bots, and can detect and block malicious traffic by redirecting it to nonroutable addresses. However, the sinkhole can also be used maliciously to redirect unwary users to unwanted IP addresses and domains. A DNS blackhole is similar; it can be used to identify domains used by spammers, domains that contain malware, and so on, and block traffic to those domains. It can also be remotely triggered (known as a RTBH). A DNS blackhole list (DNSBL) is a published list of IP addresses within DNS that