

Directed Reflection Denial of Service

Steve Gibson
Gibson Research



Attack Overview

- Perhaps the most startling aspect of this attack was that the apparent source was hundreds of the Internet's "core routers", web servers belonging to yahoo.com, and even a machine with an IP resolving to "gary7.nsa.gov".
 - Cyberarmy.com
- We appeared to be under attack by hundreds of very powerful and well-connected machines.



TCP Review

- **The establishment of a TCP connection typically requires the exchange of three Internet packets between two machines in an interchange known as the TCP three-way handshake. Here's how it works:**
 - **SYN:** A TCP client (such as a web browser, ftp client, etc.) initiates a connection with a TCP server by sending a SYN packet to the server.
 - **SYN/ACK:** When a connection-requesting SYN packet is received at an 'open' TCP service port, the server's operating system replies with a connection-accepting SYN/ACK packet.
 - **ACK:** When the client receives the server's acknowledging SYN/ACK packet for the pending connection, it replies with an ACK packet.

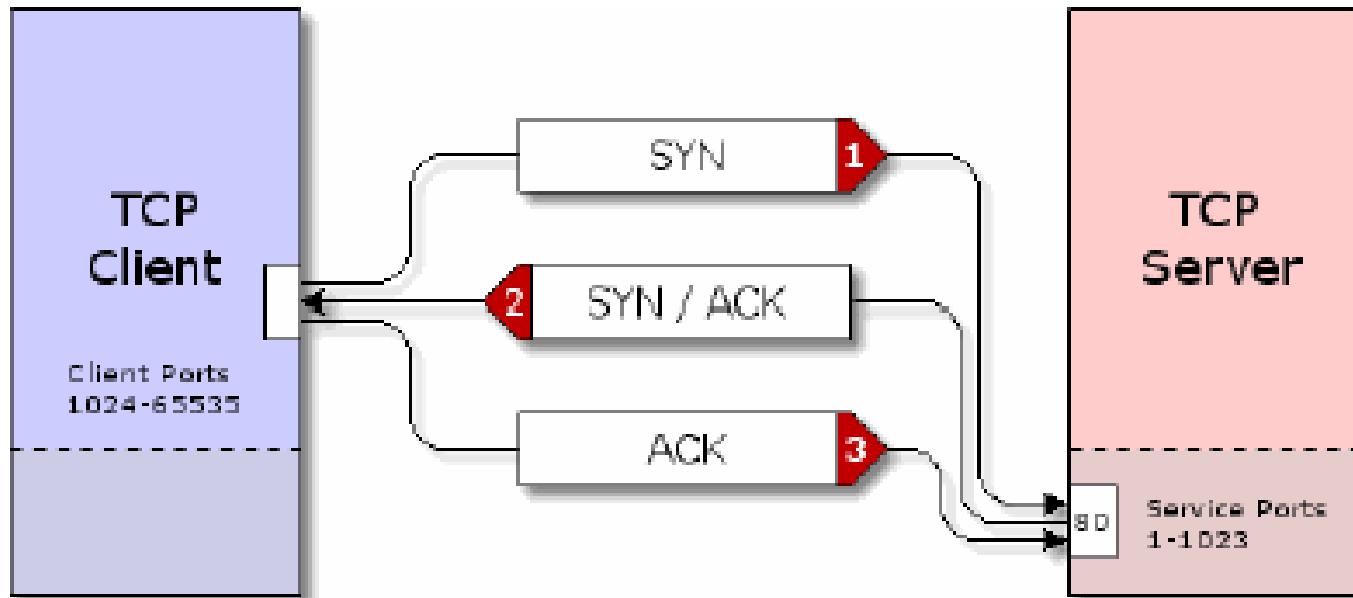


Bandwidth Consumption DoS

- **Traditional SYN flooding DoS attacks are either one-on-one**
 - (one machine sending out enough SYN packets to the target machine to effectively choke off access to the other machine)
- **or many-on-one**
 - (SYN flooding 'zombie' programs loaded by the attacker into compromised machines and commanded by the attacker to send huge volumes of SYN commands to the target machine).



Review of SYN Packets



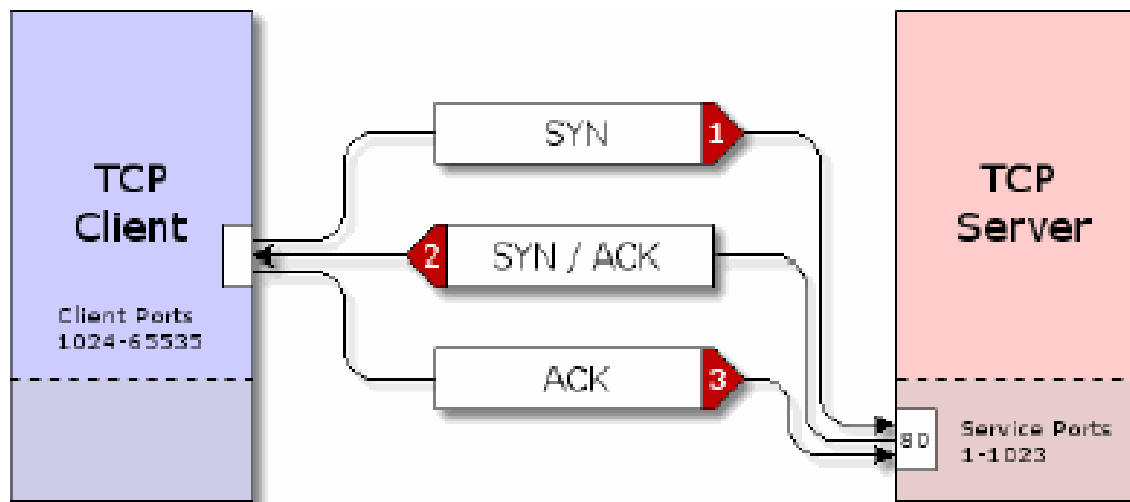
SYN: A TCP client (such as a web browser, ftp client, etc.) initiates connection with a TCP server by sending a "SYN" packet to the server.



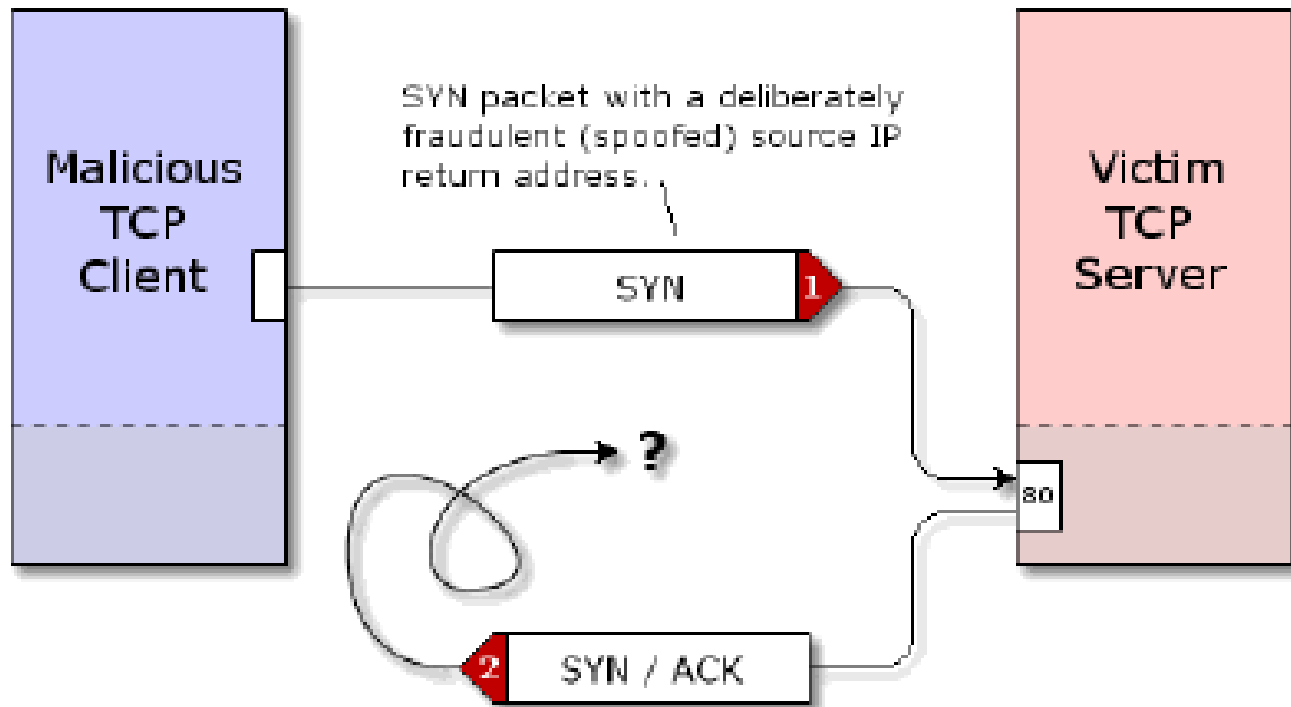
Review of SYN Packets

SYN/ACK: When a connection-requesting SYN packet is received at an "open" TCP service port, the server's operating system replies with a connection accepting the "SYN/ACK" packet.

ACK: When the client receives the server's acknowledging SYN/ACK packet for the pending connection, it replies with an ACK packet.



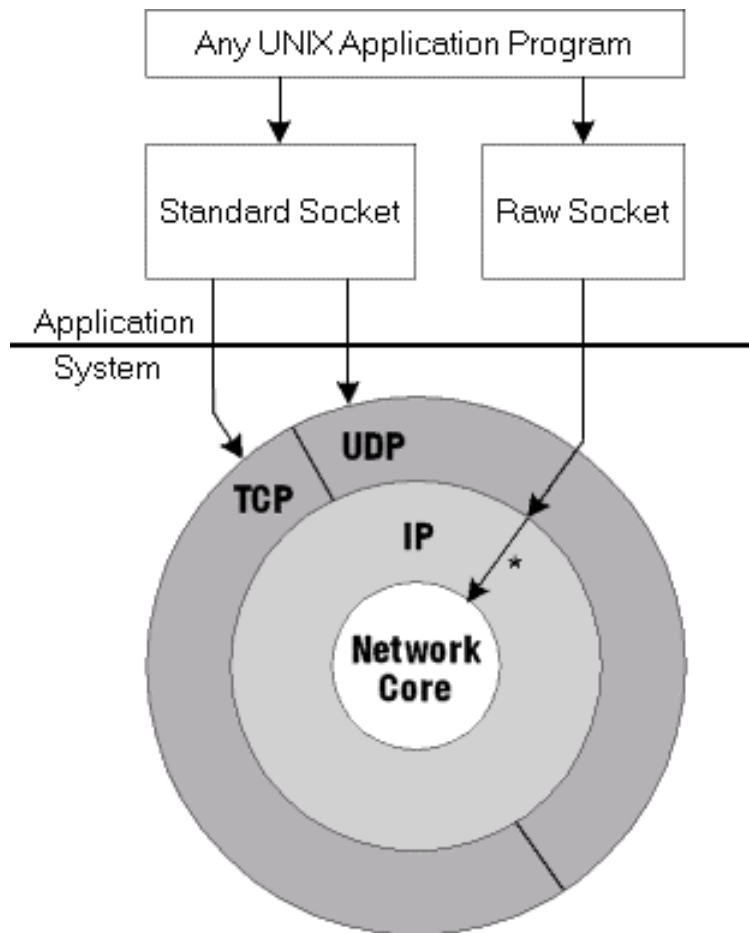
SYN Packet with Deliberately Spoofed Return Address



Through the use of "[Raw Sockets](#)", the packet's "return address" (source IP) can be overridden and falsified. When a SYN packet with a spoofed source IP arrives at the server, it appears as any other valid connection request.



Raw Socket Review



- Data is exchanged across the Internet by either establishing a bi-directional "TCP Connection" between two machines, or by sending a uni-directional "UDP Datagram" message from one machine to another. Both of these data transferring operations employ standard sockets.



Raw Sockets Review

- Smooth and orderly traffic flow across the Internet requires machines to inform each other of various non-data events such as closed ports, network congestion, unreachable IP addresses, etc. The ICMP (Internet Control Message Protocol) was created to fill this need.
- The operating system's built-in TCP/IP stack automatically and transparently generates and receives most of these "Internet plumbing" ICMP messages on behalf of the machine. To facilitate the creation of Internet plumbing applications, such as "ping" and "traceroute", which also employ ICMP messages, the Berkeley designers allowed programmers to manually generate and receive their own ICMP, and other, message traffic. As shown in the diagram, the Berkeley Sockets system provides this power through the use of a so-called "Raw Socket".
- A Raw Socket short-circuits the TCP/IP stack to open a "backdoor" directly into the underlying network data transport.
 - This provides full and direct "packet level" Internet access to any Unix sockets programmer.



SYN Packet: Destination Unknown

- The server will allocate the required memory buffers, record the information about the new connection, and send an answering SYN/ACK packet back to the client.
- But since the source IP contained in the SYN packet was deliberately falsified (it is often a random number), the SYN/ACK will be sent to a random IP address on the Internet.
- If the packet were addressed to a valid IP, the machine at that address might reply with a "RST" (reset) packet to let the server know that it did not request a connection.
- But with over 4 billion Internet addresses, the chances are that there will be no machine at the address and the packet will be discarded.



Reflection SYN Flooding

- With a reflection SYN flooding attack the attacking machines send out huge volumes of SYN packets but with the IP source address pointing to the target machine.
- The TCP three-way handshake requires that any TCP based service that receives a SYN packet must respond with a SYN/ACK packet.
- The servers and routers that receive these fraudulent SYN packets dutifully send out the SYN/ACK packet to the machine pointed to by the SYN packets IP source address.



SYN Reflector Capability

- **Consider this, any general-purpose TCP connection-accepting Internet server could be used to reflect SYN packets.**
- **Here is a short list of the more popular TCP ports:**
 - **22 (Secure Shell)**
 - **23 (Telnet)**
 - **53 (DNS)**
 - **80 (HTTP/web)**
 - **And, virtually all of the Internet's routers will accept TCP connections on port 179.**
- **To fully comprehend the potential of this new form of DoS attack consider this:**
 - **it uses a fundamental Internet communications protocol;**
 - **machines that use this protocol exist in the millions;**
 - **it is extremely easy to generate a list of 'SYN packet reflectors'.**



Generating and Using the 'SYN Packet Reflector' List

- **A simple script can be constructed to collect a large number of 'SYN packet reflection' capable routers and servers.**
 - Well-known web server farms, such as eBay and Yahoo, are easily available.
 - Simple port scans through high bandwidth IP regions will reveal thousands, if not millions, of available TCP servers.
 - Readily available tools such as *Trace Route* provide the IP address of every Internet router between the tracer and any other IP address.
- **Given a large list of SYN packet reflectors, each SYN spoofing attack host can distribute its fraudulent SYN packets evenly across every reflector on its list.**



Load Balancing the Attack

- The big win for the attacker is that since the SYN flooding machine is distributing its packets across a huge number of SYN packet reflectors, none of the innocent reflectors will experience significant levels of incomplete TCP connections.
- And, since routers generally do not retain any record of previously routed packets, it makes tracking an attack from the victim to the attacker extremely difficult.



Force Multipliers

- As if ease of attack and ubiquity of reflectors were not bad enough, it turns out that the reflectors will generate **three or four times** more SYN/ACK packets than the number of SYN packets they receive.
- Since the TCP connection that receives the SYN command is expecting to receive an ACK back from the machine it sent the SYN/ACK response to, it will send out three or four more SYN/ACK responses over the next few minutes.
- This TCP protocol feature essentially multiplies the number of malicious SYN/ACK packets being sent to the target machine by a factor of three or four.
- It also means that the flood of SYN/ACK packets will continue to disable the target site for a minute or two even after the attacker has called off the attack.



Collateral Damage

- **The basic connection unit in the Internet is the router.**
 - **Some routers serve only a small number of machines while other 'aggregation routers' collect and disperse large amounts of packet traffic from smaller networks.**
- **During normal operations, the traffic flowing through the aggregation routers can be sorted and forwarded to the router's various lower bandwidth client networks.**
- **Now imagine a SYN/ACK flood that is so large that it starts to degrade the performance of the aggregation router.**
 - **Having to process and disperse so many packets to the client networks, the router will drop and discard a portion of the packets.**
 - **Legitimate Internet clients, trying to access resources that have nothing to do with the target under attack, will also experience degraded, or complete denial of, service.**



Solutions to SYN Spoofing

- Operating system vendors responded to spoofed SYN packet DoS attacks by strengthening their TCP "protocol stacks" in various ways.
- Most of these were quantitative improvements to make their systems less vulnerable, but they did not eliminate the problem.
- Two complete, robust, and practical solutions were developed:
 - The Unix community invented a clever "stateless" TCP connection system known as "SYN-cookies".
 - Steve Gibson implemented my own different solution which was dubbed "GENESIS".
- Both of these DoS solutions arrange to stay compatible with all important aspects of the standard TCP protocol.
- They operate by eliminating all allocation of server resources after receiving a SYN packet and generating a SYN/ACK reply.

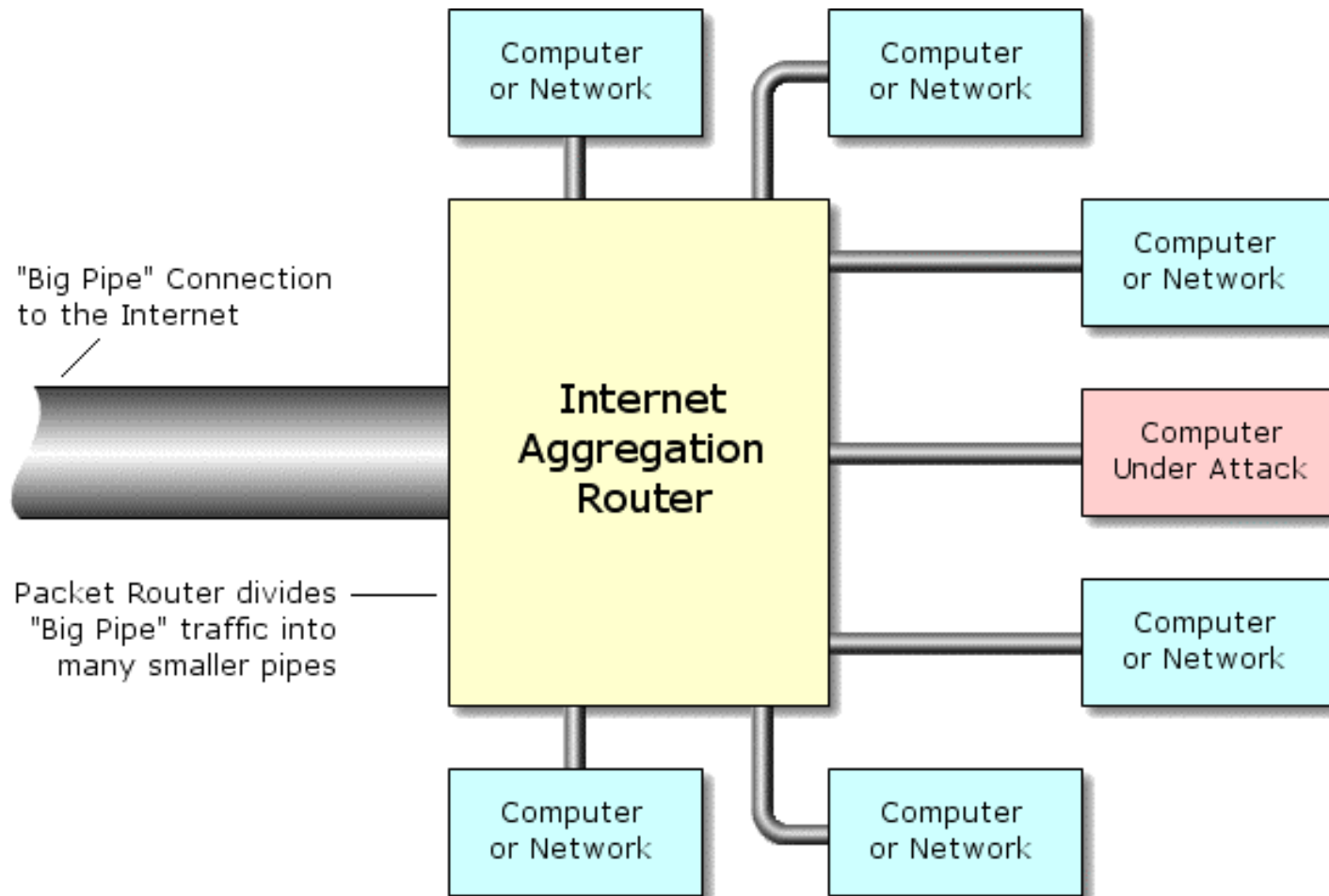


Bandwidth Consumption

- Unlike a DoS-style attack, in which a low rate of fraudulent SYN packets consumes a vulnerable server's TCP connection resources, a bandwidth attack creates a brute force flood of malicious "nonsense" Internet traffic to swamp and consume the target server's or its network connection bandwidth.
- This malicious packet flood competes with, and overwhelms, the network's valid traffic so that "good packets" have a low likelihood of surviving the flood.
- The network's servers become cut off from the rest of the Internet, and their service is denied.



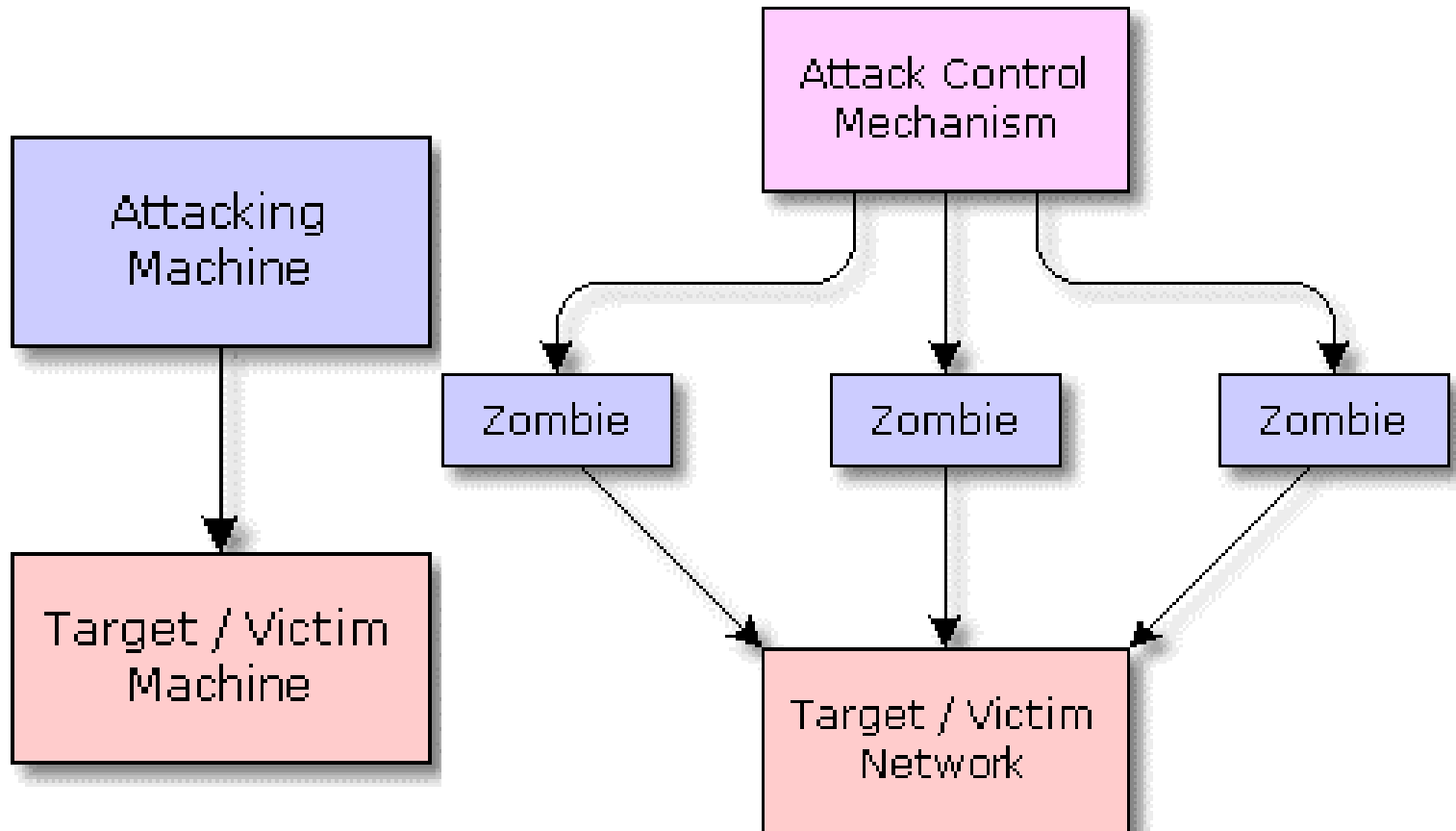
Internet Aggregation Router



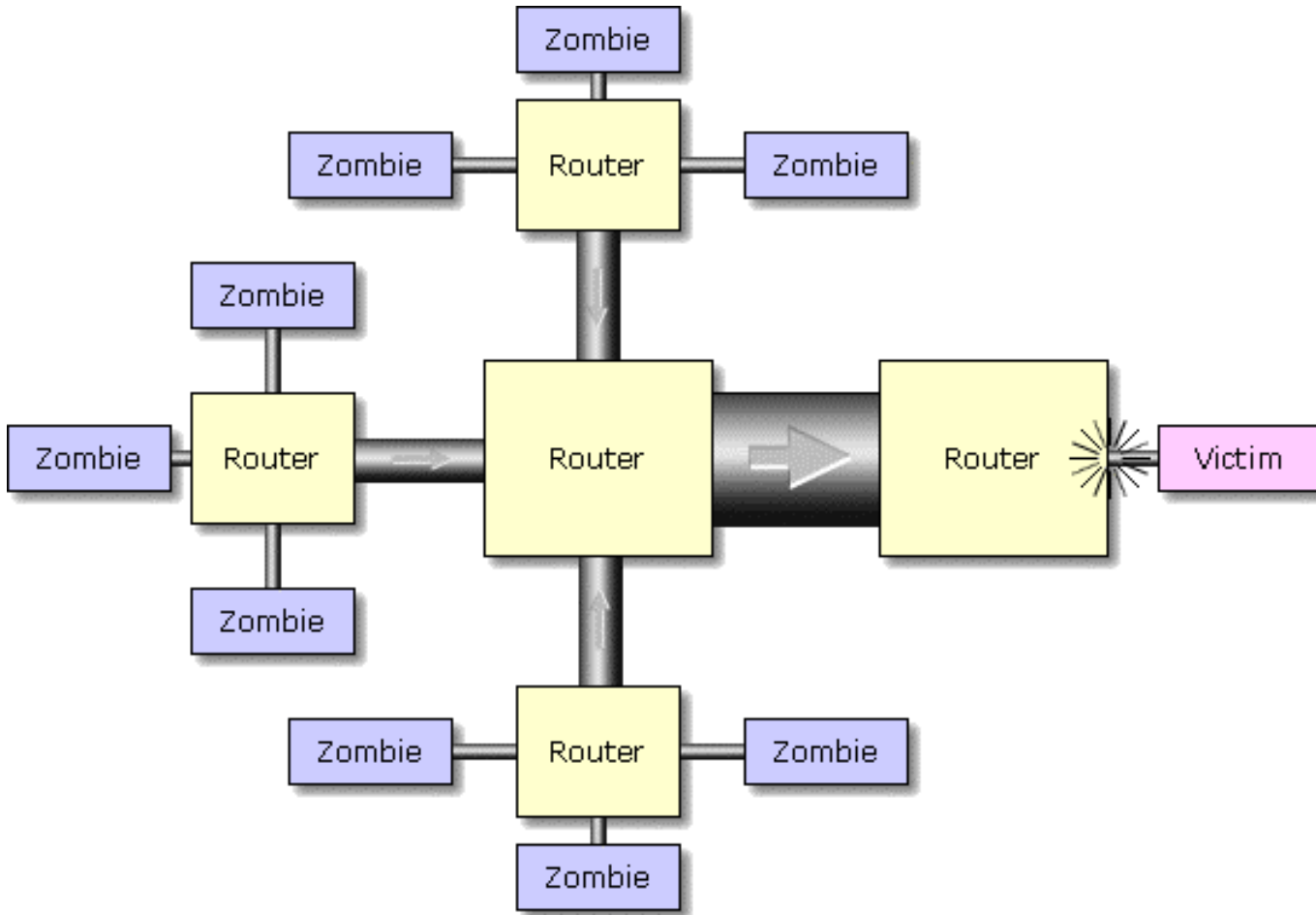
- **The computers and/or networks shown to the right are serviced by the central "aggregation router."**
 - This router is placed at the "customer edge" of the Internet service provider's network to collect and disperse traffic from many smaller customer networks.
 - Thus, many lower-bandwidth Internet connections are "aggregated" into a single high-bandwidth Internet connection for routing to the public Internet.
- **During normal operation, the traffic coming from the Internet down the "Big Pipe" will be sorted and forwarded to the router's various lower bandwidth client networks.**
- **When the Big Pipe is filled by a high volume of packets bound for just one of the router's client networks.**
 - Faced with the task of squeezing too many packets from the big pipe into the much smaller pipe, the router has no choice but to deliberately drop and discard a large percentage of the packets struggling to get through the smaller pipe.
 - Valid Internet clients, trying to access the resources on the far side of the smaller pipe, will resend their dropped packets. But these clients will generally give up after a few attempts. The victim's network is effectively blasted off the Internet by the flood of malicious traffic.



DoS versus DDoS



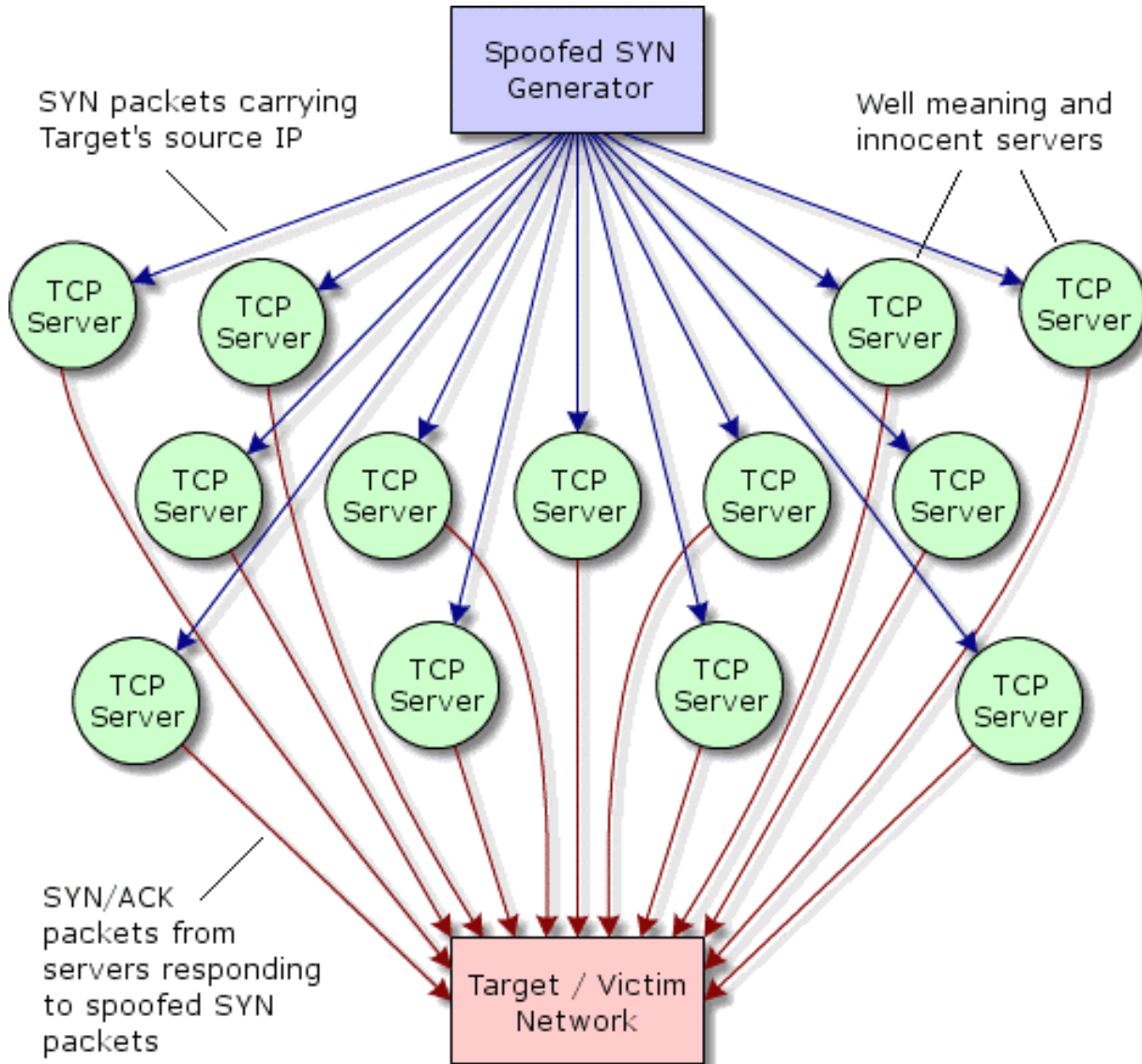
Distributed zombie traffic aggregation



- As the individual streams of traffic move across the Internet from their many separate sources, they are combined by the Internet's routers to form a single massive flood . . .



SYN FLOODING INTERNET ROUTERS (Bandwidth Attack)



- TCP servers were sending **SYN/ACK** packets to **grc.com** in the well-meaning belief that **WE** wanted to open a TCP connection with their built-in **BGP** servers.



Blocking the reflection attack

- **First, block any inbound traffic originating from the BGP service port 179.**
 - Since the malicious hacker's SYN packets were aimed at the intermediate routers' port 179, any reflected packets would be originating from that port.
 - Verio's engineer added a "filter" to the aggregation router servicing our Internet connection to block (drop) any packets inbound to us from port 179.
 - The flood of packets coming in from port 179 immediately stopped.
- **But we did NOT return to the Internet.**

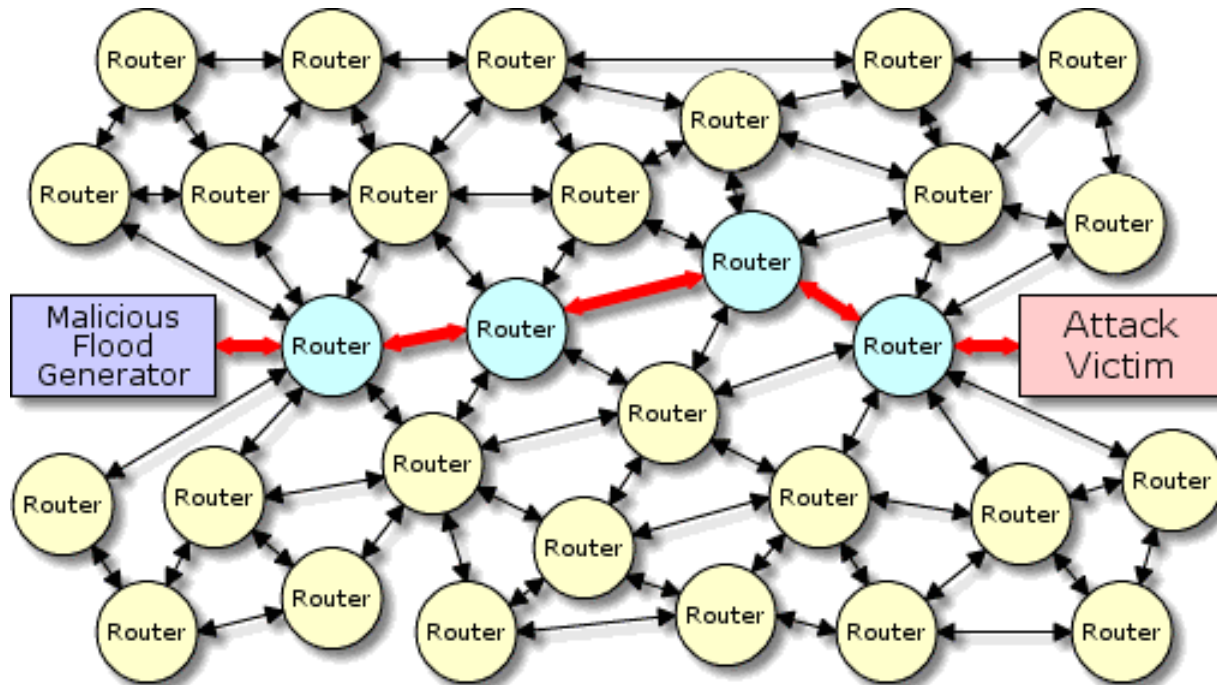


Secondary Flooding

- A fresh packet capture revealed that we were now being actively flooded by an entirely new set of Internet servers.
- Since this second set of traffic appeared only after the port 179 router traffic had been blocked, it appeared that this second wave of reflection traffic had been unable to compete with the routers' flood.
 - (You know you're in trouble when packet floods are competing to flood you.)
- With the routers traffic blocked, we were now being flooded by a SYN/ACK packets pouring in from ports 22 (Secure Shell), 23 (Telnet), 53 (DNS), and 80 (HTTP/Web).
- There were also some packets coming from port 4001 (a proxy server port) and 6668 (IRC chat).



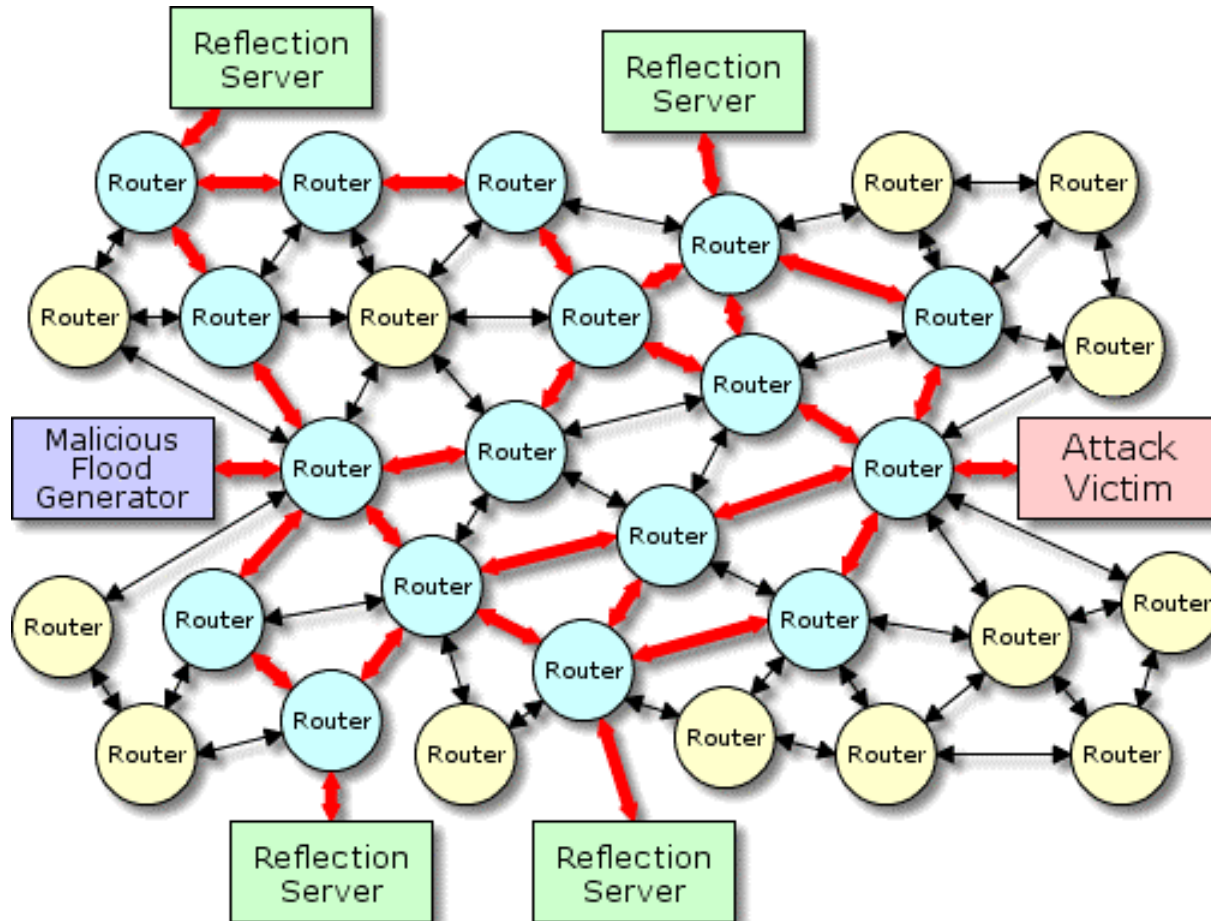
Packet Path Diffusion



- The big win for the attacker is the extreme degree of "packet path diffusion" made possible when attack traffic can be bounced off a large number of intermediate TCP servers. This diagram is a representation of the path of traffic between a single attacker and victim.

Packet Path Diffusion with Reflectors

- The addition of innocent reflection servers substantially transforms the attack.



Packet Path Diffusion with Reflectors

- Upon leaving an attacking machine, the malicious SYN packets immediately fan out.
- No longer aimed at the victim, these attack packets are instead being sent to widely spread TCP servers.
- As we know, these servers are potentially located throughout the entire Internet.
- Just a few "router hops" away from the attacker, the heavy packet flow will no longer be discernible because it will have diffused into neighboring routers rather than following a single path.



Defending against DRDoS

- Routers can be configured to filter (drop) packets destined for a particular address or group of addresses.
 - Router port 179 can be blocked as a reflector.
- Since reflected SYN/ACK packets must bounce off a TCP server, and since almost all common service ports fall within the range from 1 to 1023, blocking all inbound packets originating from the service port range will block most of the traffic being innocently generated by reflection servers. Holes in the reflection filter may have to be created to allow legitimate traffic to pass through.
- Block all inbound packets to high-numbered service ports. This has the undesirable effect that legitimate clients of the protected server could be generating connections from those blocked ports.



Defending against DRDoS

- End-user client machines cannot be protected. Most client machines spend all of their time connecting to remote servers all over the Internet and require access to data coming back from many of the most common low-numbered service ports.
- Servers could be programmed to recognize a SYN source IP address that never completes its connections and has an anomalous number of failed connections occurring within a period of time. The target of the reflection attack could be easily determined and the SYN/ACK response could be temporarily turned off.
- ISPs could prevent the transmission of fraudulently addressed packets (packets with an IP source address not within their source address space) from within their controlled networks. This control mechanism alone would have a major dampening effect on this type of attack.

