

Denial of Service (DoS)

Today: Denial of Service

ITIS 6200 / 8200

- Denial of service
 - Availability
 - Application-level DoS
 - Algorithmic complexity attacks
 - Network-level DoS
 - Distributed DoS (DDoS)
 - Amplified DoS
 - SYN flooding
 - SYN cookies
 - Defenses

Availability and Denial of Service (DoS)

ITIS 6200 / 8200

- **Availability:** Making a service on the network available for legitimate users
- **Denial of service (DoS):** An attack that disrupts availability of a service, making it unavailable for legitimate users
 - Reasons for a DoS attack
 - Competitors might DoS each other to benefit their own services
 - Criminals might DoS services unless the services pay a ransom
 - People might DoS services to make a political statement
 - Entities might DoS each other as part of warfare tactics
 - Some people might DoS for fun or revenge (e.g. online games)

DoS in the News

ITIS 6200 / 8200

Krebs on Security
In-depth security news and investigation

[Link](#)

Digital Hit Men for Hire

Brian Krebs

August 1, 2011

Cyber attacks designed to knock Web sites off line happen every day, yet shopping for a virtual hit man to launch one of these assaults has traditionally been a dicey affair. That's starting to change: Hackers are openly competing to offer services that can take out a rival online business or to settle a score.

There are dozens of underground forums where members advertise their ability to execute debilitating "distributed denial-of-service" or DDoS attacks for a price. DDoS attack services tend to charge the same prices, and **the average rate for taking a Web site offline is surprisingly affordable: about \$5 to \$10 per hour; \$40 to \$50 per day; \$350-\$400 a week; and upwards of \$1,200 per month.**

DoS in the News

ITIS 6200 / 8200

COMPUTERWORLD

[Link](#)

Extortion via DDoS on the rise

Denise Pappalardo and Ellen Messmer

May 16, 2005

Criminals are increasingly targeting corporations with distributed denial-of-service (DDoS) attacks designed not to disrupt business networks but to be used as tools to extort thousands of dollars from the companies.

Those targeted are increasingly deciding to pay the extortionists rather than accept the consequences, experts say. While reports of this type of crime have circulated for several years, most victimized companies remain reluctant to acknowledge the attacks or enlist the help of law enforcement, resulting in limited awareness of the problem and few prosecutions.

DoS in the News

ITIS 6200 / 8200



[Link](#)

DDoS makes a phishing e-mail look real

Munir Kotadia

November 8, 2006

Just as Internet users learn that clicking on a link in an e-mail purporting to come from their bank is a bad idea, phishers seem to be developing a new tactic -- launch a DDoS attack on the Web site of the company whose customers they are targeting and then send e-mails "explaining" the outage and offering an "alternative" URL.

DoS Attacks: Strategies

ITIS 6200 / 8200

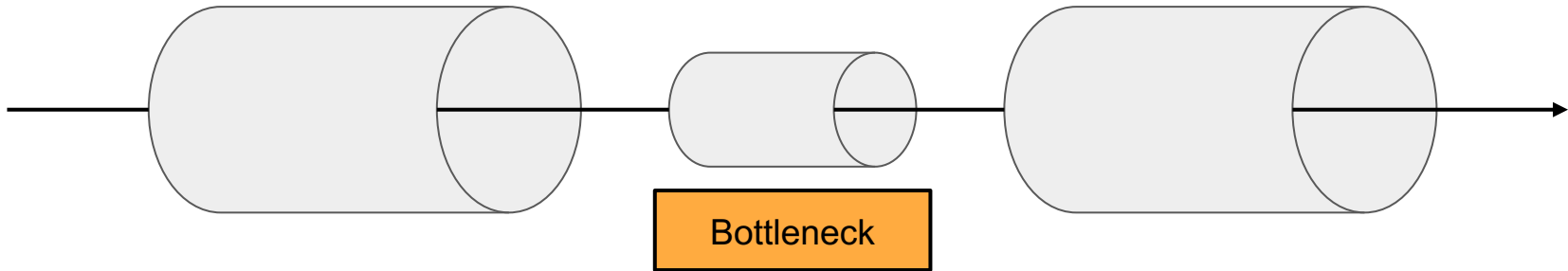
- Exploiting program flaws
 - Software vulnerabilities can cause a service to go offline
 - Example: Exploit a buffer overflow to execute a shutdown command to the system
 - Example: Exploit a SQL injection vulnerability to delete the database
- Resource exhaustion
 - Everything on the network has limited resources
 - The attacker consumes all the limited resources so legitimate users can't use them

DoS Attacks: Strategies

ITIS 6200 / 8200

- Bottlenecks

- Different parts of the system might have different resource limits
- The attacker only needs to exhaust the **bottleneck**: the part of the system with the least resources



DoS Targets

ITIS 6200 / 8200

- **Application-level DoS:** Target the high-level application running on the host
- **Network-level DoS:** Target network protocols to affect the host's Internet

Application-Level DoS

Application-Level DoS

ITIS 6200 / 8200

- Target the resources that the application uses
- Exploit features of the application itself
- Some attacks rely on **asymmetry**: A small amount of input from the attack results in a large amount of consumed resources!

Resource Consumption

ITIS 6200 / 8200

- Idea: Force the server to consume all its resources

```
int fd = open('/tmp/junk');  
char buf[4096]  
while (1) { write(fd, buf, 4096) };
```

```
while (1) { malloc(1000000000); }
```

```
while (1) { fork(); }
```

```
while (1) {  
    int fd = open(random_file());  
    write(fd, "abcde", 5);  
    close(fd);  
}
```

Exhausts filesystem space

Exhausts RAM

Exhausts processing threads

Exhausts disk I/O operations

Algorithmic Complexity Attacks

ITIS 6200 / 8200

- Consider an application that runs a sort on user-chosen data
 - What if the attacker intentionally chooses inputs that cause the worst-time runtime to occur?
- **Algorithmic complexity attack:** Supplying inputs that trigger worst-case complexity of algorithms and data structures
 - Defense: Choose algorithms with good worst-case running times
 - Mergesort is safer than quicksort against DoS!

	Expected runtime	Worst-case runtime
Mergesort	$O(n \log n)$	$O(n \log n)$
Quicksort	$O(n \log n)$	$O(n^2)$

Application-Level DoS: Defenses

ITIS 6200 / 8200

- **Identification:** Step 0 of any defense
 - You must be able to distinguish requests from different users before you can do anything else!
 - Requires some method to identify/authenticate users
 - Authenticating users might be expensive and itself vulnerable to DoS
- **Isolation:** Ensure that one user's actions do not affect another user's experience
- **Quotas:** Ensure that users can only access a certain proportion resources
 - Example: Only trusted users can execute expensive requests
 - Example: Limit each user to 4 GB of RAM and 2 CPU cores

Application-Level DoS: Defenses

ITIS 6200 / 8200

- **Proof-of-work:** Force users to spend some resources to issue a request
 - Idea: Make a DoS attack more expensive for the attacker, who now needs to spend resources
 - Example: Add a CAPTCHA, which the attacker will now have to solve (or pay for solving services)
- **Overprovisioning:** Allocate a huge amount of resources
 - Can cost the server a lot of money!
 - Often the most effective defense (“security is economics”)
 - **Content delivery network (CDN):** A service that allocates a huge amount of resources for you
 - Example of a CDN: Cloudflare
 - Cloudflare runs your service for you with a huge amount of resources

Network-Level DoS

Network-Level DoS

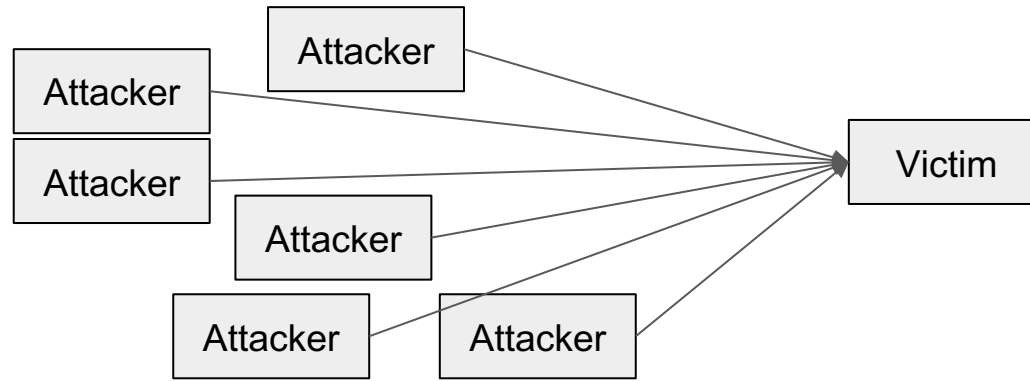
ITIS 6200 / 8200

- Approaches target network protocols to affect the victim's Internet access
 - Example: Send a huge amount of packets to the victim
- Overwhelm the victim's **bandwidth** (amount of data it can upload/download in a given time)
 - Example: The server can only upload/download 10 MB/s. The attacker sends the server 20 MB/s.
 - Lots of maximum-sized packets
- Overwhelm the victim's **packet processing capacity**
 - Example: The server can process 10 packets/second. The attacker sends the server 20 packets/second.
 - Lots of minimum-sized packets

Distributed Denial-of-Service (DDoS)

ITIS 6200 / 8200

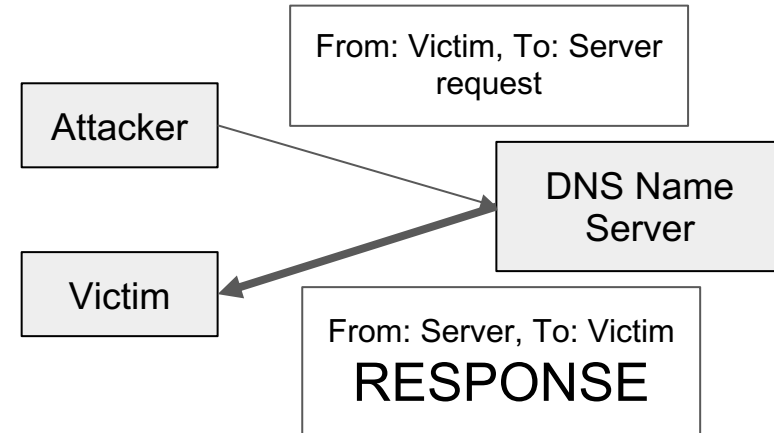
- **Distributed denial-of-service (DDoS):** Use multiple systems to overwhelm the target system
 - Controlling many systems gives the attacker a huge amount of bandwidth
 - Sending packets from many sources makes it hard for packet filters to distinguish DDoS traffic from normal traffic
 - **Botnet:** A collection of compromised computers controlled by one attacker
 - The attacker can tell all the computers on the botnet to flood a given target



Amplified Denial-of-Service

ITIS 6200 / 8200

- **Amplified denial-of-service:** Use an amplifier to overwhelm the target more effectively
 - Idea: Some services send a large response when sent a small request
 - Spoofing a small request that appears to come from the victim results in a large amount of data sent to the victim
 - Example: DNS amplification
 - Requests contain only the question
 - Responses contain answer records, authority records, and additional records



Amplified Denial-of-Service

ITIS 6200 / 8200

- **Benefits:**
 - The attacker's identity is concealed because the packets come from the amplification server
 - The attacker is able to overwhelm more bandwidth with relatively little bandwidth
 - Amplification servers often have massive bandwidths to support large numbers of users
- **Drawbacks:**
 - Requires blind spoofing capability
 - Cannot work over TCP, since TCP spoofing is assumed to be hard, only UDP protocols

Network-Level DoS: Defenses

ITIS 6200 / 8200

- **Packet filter:** Discard any packets that are part of the DoS attack
 - Discard packets where the source IP is the attacker's IP address
 - Find some pattern in the content of the DoS packets to distinguish DoS packets from legitimate packets
 - The packet filter must be before the bottleneck
- **Subverting packet filters**
 - Spoof DoS packets so that packets look like they're coming from many IP addresses
 - Packet filters can't use IP addresses to filter packets anymore!
 - Hard to defend against
 - Rely on anti-spoofing mechanisms on the network
 - Distributed DoS actually send packets from many IP addresses
 - Packet filters need to be much more sophisticated to defend against DDoS attacks

Network-Level DoS: Defenses

ITIS 6200 / 8200

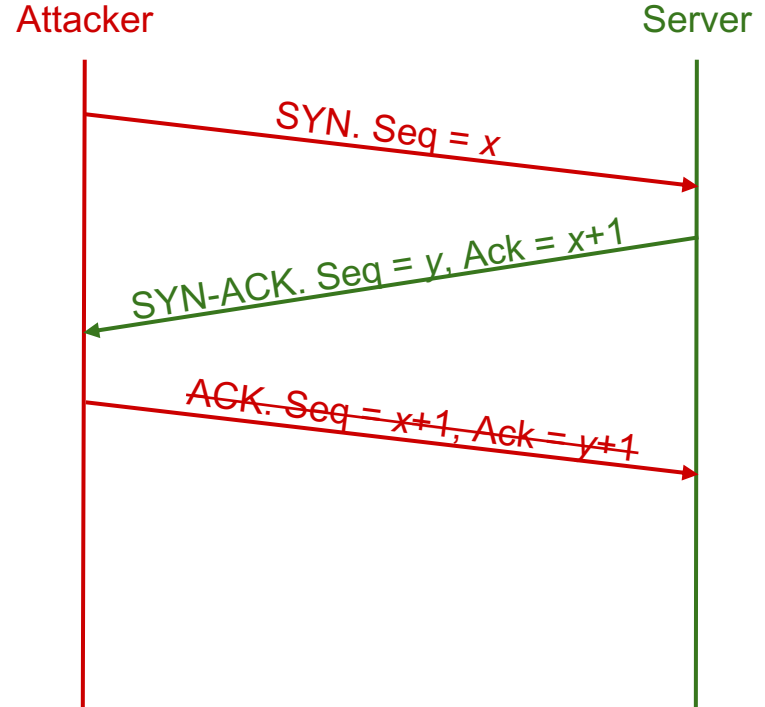
- **Overprovisioning:** Purchase enough networking bandwidth and equipment to make it harder for attackers to overwhelm the network
 - Again, depends on your threat model

SYN Flooding and SYN Cookies

SYN Flooding

ITIS 6200 / 8200

- A type of DoS that exploits many TCP connections
 - Each connection established by the server needs to allocate some memory
 - Used to store sequence numbers, ACK numbers, buffered data, etc.
 - Idea: Establish many connections with the server, causing it to consume a lot of memory
- TCP state is allocated upon receiving a SYN
 - The attacker only needs to send the SYN, so the attacker doesn't consume its own resources!



SYN Flooding: Defenses

ITIS 6200 / 8200

- **Overprovisioning:** Ensure the server has a lot of memory
 - Can be expensive and depends on your threat model
- **Filtering:** Ensure that only legitimate connections will create state
 - Same problems as standard packet filtering for network-level DoS attacks
 - Hard to distinguish legitimate traffic so early in the connection
 - Attacker can spoof source address since they only need to send the SYN, not the ACK
- **SYN cookies:** Don't store state!
 - Relies on the client to store the server's state
 - The client returns the state to the server in the ACK packet of the handshake

Idealized SYN Cookies

ITIS 6200 / 8200

Client

Server

SYN. Seq = x

SYN-ACK. Seq = y . Ack = $x+1$. $\langle \text{State} \rangle$

ACK. Seq = $x+1$, Ack = $y+1$. $\langle \text{State} \rangle$. Data

The server generates state for the client but *doesn't* save it, sending it to the client instead encoded with a secret

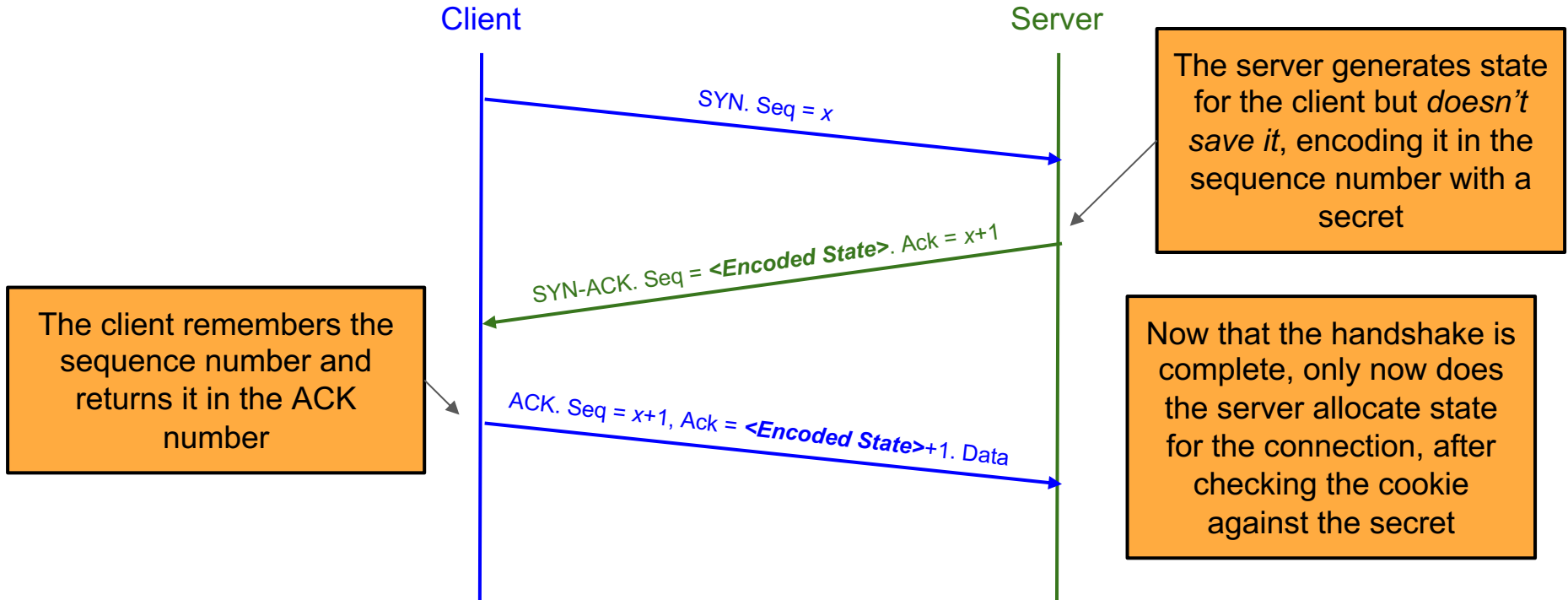
The client stores the state on behalf of the server and returns it in the ACK packet

Now that the handshake is complete, only now does the server allocate state for the connection, after checking the cookie against the secret

Issue: TCP doesn't have a mechanism to store state! What field of the SYN-ACK packet could we store data in?

Practical SYN Cookies

ITIS 6200 / 8200



Practical SYN Cookies

ITIS 6200 / 8200

- Observation: The server doesn't create state until the handshake is completed, so the attacker can't spoof source addresses
 - Filtering becomes easier with SYN cookies
- We can generalize this: Instead of holding state in the server, encode it with a secret and send it to the client, who will return it when it is next needed
 - Requires enough bits to encode the state
 - We must make sure that checking the state against the secret is inexpensive, or this becomes another DoS vector!

Summary: Denial of Service

ITIS 6200 / 8200

- **Availability:** Making sure users are able to use a service
 - DoS attacks availability of services
- **Application-level DoS:** Attacks the high-level applications
 - Algorithmic complexity attacks: Attack using inputs that cause the worst-case runtime of an algorithm
 - Defense: Identification, isolation, and quotas
 - Defense: Proof of work
- **Network-level DoS:** Attacks the network of a service
 - Typically floods either the network bandwidth or the packet processing capacity
 - Distributed DoS: Use multiple computers to flood a network at the same time
 - Amplified DoS: Use an amplifier to turn a small input into a large output, spoofing packets so the reply goes to the victim
 - Defense: Packet filtering
- All DoS attacks can be defended against by overprovisioning