Managing the Threat of Denial-of-Service Attacks

CERT® Coordination Center

Allen Householder, CERT/CC
Art Manion, CERT/CC
Linda Pesante, CERT/CC
George M. Weaver, CERT/CC

In collaboration with:
Rob Thomas

v10.0
October 2001

1. Introduction

Denial-of-service (DoS) attacks have been launched against Internet sites for years. They are a significant problem because they can shut an organization off from the Internet and because there is no comprehensive solution—no silver bullet—for protecting your site or recovering from a denial of service. In this paper, we will describe the current situation with denial-of-service attacks and explore ways of addressing the problem.

Much has been written previously about denial of service. In fact, the CERT Coordination Center (CERT/CC) published alerts related to these attacks as early as 1996. CERT/CC publications and other references can be found at the end of this paper. In November 1999, the CERT/CC hosted a workshop in which 30 experts from around the world addressed the increasing sophistication of DoS tools to launch distributed denial-of-service (DDoS) attacks. The results of that workshop are published on the CERT/CC web site (http://www.cert.org/reports/dsit_workshop.pdf). (For additional publications, see the *References* section of this paper.) The report contained the most current knowledge of denial of service at the time. This paper provides the knowledge gained since the workshop. The information we provide is geared to commercial business. Though Internet service providers and home users can benefit, we have not tailored this paper to their particular needs.

The next section provides background information about denial-of-service attacks. It is followed by information on steps you can take to reduce your risk of attack as well as how to identify attacks when they happen and respond to them. Finally, we take a look at future possibilities. An appendix contains additional information about denial of service.

2. Background: Insight into Denial of Service

Denial of service is accomplished technologically—the primary goal of an attack is to deny the victim(s) access to a particular resource. It is an explicit attempt by attackers to prevent legitimate users of a computer-related service from using that service. But, as any information and network security issue, combating denial of service is primarily an exercise in risk management. To mitigate the risk, you need to make business decisions as well as technical decisions.

Managing the risks posed by denials of service requires a multi-pronged approach:

- Design your business for survivability. Have business continuity provisions in place.

- Design your network for survivability. Take steps that help to ensure that critical services continue in spite of attacks or failures. (Keep in mind that increased complexity means increased costs and decreased reliability.)

- Be a good netizen (net citizen). Your potential to be attacked depends on the security of other sites and vice versa. The threat to your network is directly proportional to the extent that other Internet users, including home users, adhere to good practices. Conversely, the threat that your network represents to others is directly proportional to the extent that your organization adheres to good practices.

Denial of service may be indistinguishable from a heavy (but otherwise legitimate) load on your network. For example, you might be flooded with legitimate connections to your web site as a result of a major news event such as the disaster that occurred on September 11, 2001. Users might have difficultly connecting to your web site simply because so many people are trying to connect at one time and not because you are the target of a denial-of-service attack. It is important to establish criteria by which you will declare your site "under attack" and invoke emergency procedures. Mitigation strategies for attacks and heavy, legitimate traffic may be similar.

Frequency and Scope

How prevalent are denial-of-service attacks in the Internet today? Researchers at the Cooperative Association for Internet Data Analysis (CAIDA) address this question in their paper, "Inferring Internet Denial-of-Service Activity." (see *References*.) Using a technique called backscatter analysis, the researchers monitored unsolicited traffic to unpopulated address space. Their theory is that DoS traffic that uses random spoofed source addresses will generate some response traffic to the entire Internet address space, including unpopulated space. Their results (February 2001): "Using backscatter analysis, we observed 12,805 attacks on over 5,000 distinct Internet hosts belonging to more than 2,000 distinct organizations during a three-week period."

In addition, CAIDA reports that 90% of attacks last for one hour or less; 90% are TCP-based attacks, and around 40% reach rates of 500 packets per second (pps) or greater. Analyzed attacks peaked at around 500,000 pps. Other anecdotal sources report larger attacks consuming 35 megabits per second (Mbps) for periods of around 72 hours, with high-volume attacks reaching 800 Mbps.

You may be wondering about your chances of being attacked. We advise you to count on it and plan for it. Identify "worst cases" but recognize that complete risk avoidance is impossible. There is no such thing as 100% uptime. Risk mitigation is a balancing act. The specific decisions and level of investment depend on your particular business priorities and risks.

Damages and Costs

There may be hidden costs associated with denial-of-service attacks. For example, the direct target of a DoS attack may not be the only victim. An attack against one site may affect network resources that serve multiple sites. Or resources you share with other parties (upstream bandwidth) may be consumed by an attack on someone else—another customer of your Internet service provider is attacked, so your upstream connections and routers are not as available to handle your legitimate traffic. Thus, even when you are not the target of an attack, you might experience increased network latency and packet loss, or possibly a complete outage.

You may have additional costs because of the need to size notification resources (such as logs, mail spools, and paging services) to absorb attack-related events. Logging systems need to cope with significant deviations in the amount of data logged during attacks.

Ideally, logging systems should use an out-of-band channel so that logging traffic does not add to the volume of DoS traffic that may be passed to the internal network. Centralized logging systems, considered a best security practice, may be stressed by receiving log data from multiple locations. Mail queues may fill up during a prolonged outage.

Network traffic generated by the attack can result in incremental bandwidth costs—when you pay per byte, you pay for the increased traffic caused by the attack. In addition, your upstream Internet provider might or might not be amenable to waiving penalty charges caused by flood traffic. It's good to know this ahead of time. Other issues that create hidden costs are insurance or legal fees or possible third-party liability resulting from your involvement in an attack.

Security Spending

Part of your organization's security posture is a function of its spending decisions. Let's assume that an organization has an operational IT infrastructure, including Internet services and connectivity, and that the organization has $1,000,000 to defend against DoS attacks. Following are just three examples of how the organization might choose to expend those funds.

Example 1

Provide excess capacity to absorb some attacks, hire experienced staff, and provide defensive network equipment.

$200,000	extra bandwidth and router throughput
$500,000	experienced, highly skilled staff
$300,000	firewall, load-balancing, traffic-shaping technology

Example 2

Distribute web services to avoid single point of attack; cover costs of attack with insurance.

$700,000	content distribution (Akamai, Digital Island, etc.)
$300,000	insurance premiums

Example 3

Retain a managed security service to handle attacks, provide some in-house staff and defenses, contract with ISP to respond quickly to attacks.

$200,000	firewall, load-balancing, traffic-shaping technology
$300,000	managed security services
$300,000	four-hour response service from ISP
$200,000	staff

This illustration demonstrates that you have a wide variety of options. When deciding on your company's response to DoS attacks, consider these questions: What are the chances of an attack hitting your company? What are the chances of an attack being of a certain type and/or magnitude? What level of risk is acceptable? How important is the Internet to your business? How long can you function without some or all Internet services? Which services?

In addition, business continuity plans should address loss of both critical and noncritical systems, though this doesn't change the services that should be prioritized as critical. Finally, because each attack has its own idiosyncrasies, you may need to extensively customize technical remedies, remaining aware that technical countermeasures are not 100 percent effective. In the next sections, we provide information that you can use as input for the decisions you need to make. The appendix contains additional, more technical background information about denial-of-service attacks.

3. Handling Denial of Service

The familiar protect-detect-react cycle is useful when considering what to do about denial-of-service attacks.

Protecting – Among the aspects of protecting your systems—and your business—are looking at network design, discussing your agreement with your ISP, putting detection mechanisms and a response plan in place, and perhaps taking out an insurance policy. Proper preparation is essential for effective detection and reaction. Unfortunately, some sites begin their cycle with detection and reaction, triggering preparation steps after a "lessons learned" experience.

Detecting – Your ability to detect attacks directly affects your ability to react appropriately and to limit damages. Among the approaches you can take are instituting procedures for analyzing logs and using automated intrusion detection systems.

Reacting – Reaction steps, hopefully put in place as part of preparing for an attack, include following your response plan, implementing specific steps based on the type of attack, calling your ISP, enabling backup links, moving content, and more. Technical steps include traffic limiting, blocking, and filtering.

In the following sections, we discuss each of these three topics in more detail, and describe the preparations and actions you can take to manage the risks posed by denial-of-service attacks. You can find more information in the CERT security practices in *Detecting Signs of Intrusions* and *Responding to Intrusions*,[1] and in the training we provide for computer security incident response teams (CSIRTs) and their managers.

Protect Your Systems and Prepare for Attacks

Developing defenses against denial-of-service attacks only makes sense if you first know what you are trying to defend. Since it is not likely to be possible to defend all your systems against all possible DoS attacks, you need to make well-reasoned tradeoffs when deciding where to expend limited resources in building defenses. There are various ways to do this. The SEI has developed an information security risk evaluation called OCTAVE$^{SM)}$[2], which helps organizations identify vulnerabilities and threats to their

[1] CERT security practices can be found online at www.cert.org/security-improvement and the book, CERT *Guide to Network and Security Practices*, written by Julia Allen and published by Addison-Wesley.

[2] Operationally Critical Threat, Asset, and Vulnerability Evaluation, described further at www.cert.org/octave/.

critical information and plan protection strategies, so we won't go into great detail here. Rather, we offer some general principles that apply to DoS defense.

First, as with other information security decisions, what you do, when, and how, are rooted in business risk management. Careful analysis of your business strategy and objectives can lead to the identification of mission-critical technology and information assets, that is, what you need to protect. Differentiating critical services from noncritical services is one of the key results of this analysis. This information can then be used to establish priorities for the strategic and tactical tradeoffs that are necessary in the event of a DoS attack.

It is helpful to note that any of the techniques you might use to protect against DoS attacks are generally useful in building scalable networks and systems. As we mentioned earlier, a denial-of-service attack may be indistinguishable from legitimate "flash crowd" traffic.

It is also important to identify and understand the interdependencies of various services provided on your network. Any service is at least as critical as the most important service depending on it. For example, if your organization determines that web (HTTP) and email (SMTP) services are business-critical, then it follows that the Domain Name System (DNS) is also critical since both web and email services need DNS in order to function properly.

In general, systems and networks can be engineered to respond to a DoS attack by doing one of these things:

- Absorb the attack. This implies that additional capacity has already been planned for, installed, and tested before an attack begins. On the negative side, there is an additional resource cost for this excess capacity even when no attacks are currently under way.
- Degrade services. Once the critical services have been identified, it may be possible to design the network, systems, and applications in such a way that noncritical services can be degraded in favor of keeping critical services functional through an attack. If the attack is protracted or extremely heavy, it may become necessary to completely disable noncritical services to provide additional capacity to critical services.
- Shut down services. It is plausible that an organization could decide to simply shut down all services until an attack has subsided. While certainly not an optimal choice, it may be a reasonable response for some.

Your reaction to a DoS attack depends a great deal on the preparations made before an attack. Once an attack is under way, it may be too late to configure and install additional capacity or monitoring. These need to be in place ahead of time. It is also important to have communication plans in place. It is critical during and after an attack to effectively communicate both within an organization (among the technical folks, with senior management, etc.) as well as outside an organization (with service providers, possibly law enforcement, media, and others).

Protecting a network and systems from DoS attacks centers around three topics:

a) Designing the network and systems for survivability

b) Monitoring ongoing operations – knowing what's "normal" for your network so that you can detect changes to this normal behavior

c) Preparing the organization in nontechnical ways so personnel are prepared to react effectively

a. Designing for Survivability

Survivability is the ability of a network computing system to provide essential services in the presence of attacks and failures, and to recover full services in a timely manner. The main objective of designing networks and systems to protect against DoS attacks is to keep critical services operational as long as possible during an attack. In some cases, that may not be enough, since the attacker may be able to completely overwhelm your network's capacity. In that event, however, the issue becomes a business continuity issue rather than a technical one. In this paper, we focus on the technical issues by discussing general principles that apply to the survivability objective:

- Separate, or compartmentalize, critical services wherever practical. Grandma knew what she was talking about when she said, "Don't put all your eggs in one basket."
- Overprovision as much as possible. Have more capacity than you need on a typical day to the extent permitted by your constraints of time, money, resources, or complexity.
- Minimize your "target cross-section." In military aircraft, stealth technology reduces the radar reflections that attackers can use to aim their anti-aircraft weaponry. Similarly, a well-implemented network can present a small target to attackers by limiting publicly visible systems and services to the minimum required to meet the business needs of the organization.

The sections below contain further discussion of each principle, including their benefits and drawbacks.

Separate critical services

The separation of critical services from noncritical services can occur on many levels. At the physical layer, you might use redundant links, each connected to different ISPs via different carrier networks routed through diverse paths. Additionally, you can gain additional redundancy by using multiple data centers or hosting locations.

At the link layer, separation could involve the isolation of broadcast and/or collision domains through the use of multiple hubs, switches, VLANs or ELANs. This helps reduce the risk of "ARP storms" during heavy scanning events or DoS attacks. (ARP is the Address Resolution Protocol, which is used by computers to resolve hardware addresses from IP addresses. ARP requests are legitimate, but during denial-of-service attacks, their volume can become so massive that your system can be negatively affected.) At the network layer, IP subnetting allows for increased control of traffic between systems, thereby limiting the damage an attack on one system or subnet can do to another. Subnetting can also help isolate publicly visible systems and services from internal systems by establishing a DMZ (demilitarized zone) between the external and internal networks.

Further separation can be achieved by using distinct servers for different services. There are many possible ways to do this, but some general approaches are

- Separating public services from private services
- Splitting Internet, extranet, and intranet services
- Dividing n-tier architectures into their components: web servers, application servers, database servers, and so on
- Using single-purpose servers for each service (SMTP, HTTP, FTP, DNS, and others)

Even within a server, it is possible to segregate resources that are likely to come into contention. For example, you can use separate file systems for the operating system, user areas, and log files.

Overprovision capacity where possible

To keep your systems operating during a period of heavy load, you need more computing capacity than your organization normally requires. You need to make provisions for sudden surges in network traffic, regardless of their source. At this level of preparation, it is not significant whether those surges are caused by malicious behavior (DoS attack) or legitimate external events (for example, news sites experience surges during major events, corporate sites after big announcements, product sites after a Superbowl commercial, email servers during the holiday e-card season). Extra capacity may take many forms—bandwidth, memory, processor speed, TCP connection buffers, and other resources.

In evaluating capacity considerations in preparation for traffic surges, recall that most network devices and computer systems are limited not by raw bandwidth capacity (bits per second) but rather by their packet processing ability (packets per second). Many devices that operate at wire speed during normal[3] operation may experience dramatic degradation of performance when forced to deal with large volumes of small packets. Attackers know this, and as a result many DoS agents use small packets in their attacks.

As we noted earlier, careful analysis of your network infrastructure and critical services is essential. When planning additional capacity, it is also important to assess the impact that capacity will have on all the other systems that make up your critical services. There is a distinct risk that adding capacity to one part of the network may just expose a bottleneck elsewhere, or even have a cascade effect. For example, upgrading a T-1 to a DS-3 link might mean that your gateway router and firewall will be undersized. But once you

[3] In this instance, by “normal” we mean to imply “traffic that exhibits a reasonable distribution of packet sizes.”

upgrade the network components, you may discover that the servers (or worse, your applications) cannot cope with the increased load. Thus, the realized cost of the link upgrade expands to become much more than just ISP and carrier fees.

Another risk when increasing capacity is that new bottlenecks may be introduced. Imagine a web-hosting environment where new servers are added to an existing site in conjunction with a load-balancing appliance. The load balancer constitutes a new potential bottleneck and must therefore be sized appropriately. A hidden bottleneck might not be technological in origin at all—the additional complexity involved in configuring and managing all that extra capacity might strain already harried system administrators and network engineers.

Minimize your target cross-section

Reducing the target you present a potential attacker can be achieved through a number of methods. The primary objectives are to (1) present a small initial target and (2) limit the damage that an attack on that target can have.

A few examples of possible approaches include these:

- Disable unneeded services. As an example of the "principle of least privilege," all services that are not expressly required for business operations should be disabled. Many operating systems have numerous services enabled by default, unnecessarily exposing networks to attacks that aren't even related to the particular service they intend (or need) to provide.

- Hide the internals of your network. In many situations, there is little need for external users to be able to gather information about internal network configurations. Use of split-DNS, Network Address Translation (NAT), and blocking ICMP messages at the network edges can be effective methods to reduce the leakage of internal configuration details to the outside world.

- Filter[4] all non-essential traffic as close to the source as possible. By dropping unneeded traffic as early as possible in a network (through the use of ingress and egress filtering), the impact of a DoS attack may be limited to the edges of the network. When done properly, traffic filtering can help protect your systems from being overloaded by intercepting the attack upstream, and it can limit the risk of attackers using your systems[5] for an attack on another site.

  The use of filtering is not without its downside, however. To be able to filter traffic, the filter must be along the path that traffic will take. This may result in

[4] By "filtering" we mean the general ability to selectively permit or deny traffic into, out of, or through a network device or server. In this context, filtering may occur at many different levels, for example: packet filtering, stateful inspection, application proxy, or content filtering.

[5] A specific example of filtering to limit attackers is to block IRC (6667/TCP) traffic outbound from your network. Many DoS agents open outbound IRC connections to connect to control channels on external networks. Blocking this communication can severely limit the usefulness of agents on your network to attackers. (It is always possible for an attacker to use a nonstandard port for similar purposes, so this is not entirely foolproof.)

sub-optimal routing decisions in some networks. Filtering can also lead to performance bottlenecks in the network if the filtering devices are pushed beyond their capacity. Additionally, the complexity involved in implementing high-quality filtering can make subsequent maintenance and troubleshooting quite difficult.

- Load-balance across multiple servers and/or sites – Load balancing at a single site can be effective at dissipating attacks that take advantage of host resource starvation issues, but single-site solutions are still susceptible to attacks against network resources such as bandwidth. Multi-site load balancing may improve survivability in the face of network-based attacks, but there is a possibility of significantly increased operational complexity at significantly higher cost. Content distribution networks can help by effectively caching content further downstream, but the relatively high cost of entry into most content distribution solutions can be prohibitive to all but those with the largest budgets.

  Load balancing solutions can lead to other technical issues surrounding content, application, and data replication that are not always easy to solve. Applications may need to be rewritten for load-balanced systems to accommodate the introduction of resource contention among multiple instances. Real-time data replication across multiple sites is neither easy nor cheap, and the cost of having all that extra hardware available when it's not needed can be a drain on scarce resources.

- Minimize internal servers' dependency on external services. So, you've split all your externally visible services from your internal servers and have filtered everything you can. But what happens to your internal email systems if your DNS servers can't resolve anything outside your network? Even if you are not the direct target of an attack, your internal users' ability to get to the Internet may be impeded by attacks upstream from your network. Understanding the interdependency of internal services and external services in fulfilling their mission is an essential part of designing survivable systems.

- Use multiple operating systems to create "biodiversity." Most worms, viruses, and DoS tools target specific operating systems. Using multiple operating systems may aid survivability in the event of an operating-system-specific attack. On the other hand, more diversity leads to more complexity, which in turn can lead to increased costs for configuration & maintenance expertise and possibly licensing costs.

b. Monitoring ongoing operations

To be able to detect anomalous behavior, you first must be able to characterize what "normal" means in the context of your network. Things to look for include system and network performance metrics, network protocol mix, and network traffic flows. Establishing a baseline of normal traffic patterns on your network allows your system administrators to quickly identify usual behavior such as that seen during DoS attacks and traffic surges.

When planning the monitoring capabilities in your network, it makes sense to focus your attention on the resources most frequently consumed in attacks and surges. For example, network monitoring might consist of throughput measurements such as bandwidth use and packet rates, as well as device performance metrics like router CPU and memory utilization or switch backplane utilization.

Similarly, host-level monitoring could include gathering performance statistics like CPU and memory utilization or availability of free disk space, along with network behavior such as the number of TCP connections in particular states (*syn_wait*, *fin_wait*, *established*, etc.) or interface utilization. Automating routine tasks, such as backups and log rotation, can also free up administrators' time for analysis and improve their ability to respond to an attack.

Intrusion detection systems can augment performance statistics with the ability to recognize potential attack patterns in network traffic. However, they might detect abrupt traffic increases as attacks. Such "false positives" may still be useful, though, because you'll at least know that something is going on that requires your attention even if it's not malicious in nature.

On all but the smallest networks, it is probably not feasible to monitor every single host, link, or network device. The ability to generate logs far outstrips most organizations' abilities to store them, let alone analyze them. Therefore, priority should be given to core services such as backbone links, major servers, core routers and switches, and gateway firewalls. Again, the identification of critical services and interdependencies gives you a way to prioritize monitoring choices.

One last note about system and network monitoring: it is important to use caution when implementing remote monitoring capabilities so that you don't create new bottlenecks. For instance, let's say you've configured your Internet gateway router to log every dropped packet to a central logging system. Under normal conditions, this provides a convenient collection capability that can be leveraged for baseline measurements and analysis of intrusion attempts. However, in the case of a packet flooding DoS attack, it is possible that even though the router is capable of dropping every attack packet, it will consume all its available computing resources in generating log packets to send to the collection station. Furthermore, if the collection station is receiving log events from multiple sources, that system (or the network between the machine generating the logs and the one receiving them) may also become overloaded if one or more devices start generating massive amounts of log information at the same time. This can happen if multiple Internet links are hit by a DoS attack simultaneously, or if the collection station is already operating near peak capacity. The cascading impact of a DoS attack doesn't stop there, though. The CERT/CC has received reports indicating that even if the devices generating and receiving the logs and the networks in between survive the increased load, automated system backups may fail during or after an attack due to a combination of network load and dramatically increased log file sizes.

c. Preparing the organization to react

In addition to the various network and system design guidelines presented above, you can take nontechnical steps to mitigate the risks posed by DoS attacks. Among these are

- Cultivate an analysis capability
- Create an incident response plan (and allocate resources to its maintenance and execution)
- Develop an ongoing relationship with your upstream provider(s)

Cultivate an analysis capability

Merely gathering statistics and collecting logs is of little value if you are unable to analyze and interpret them in a timely manner. Automated tools can help reduce the vast amounts of information, but a trained eye is still required to discern the relevant from the banal. As mentioned above, it is not always easy to distinguish malicious DoS traffic from a heavy but otherwise legitimate load.

Some organizations might choose to develop this capability in-house, while others might outsource it. The decision of where to house this function is best made in consideration of the frequency and severity of incidents that require this level of expertise. Many organizations will find that there is insufficient demand for a full-time team in-house, and will choose to use consulting services when needed for tactical purposes.

A third option is to take advantage of public analysis services such as mailing lists or newsgroups which focus on security incident analysis and response. This the option has the lowest associated cost, but it may not meet the privacy concerns or timeliness requirements of some organizations.

Create an incident response plan

An incident response plan is vital to the successful handling of any incident. Here are examples of the recommendations that the CERT/CC includes in its training for computer security incident response teams:

- Document standard operating procedures that include information on how to proceed during an incident, how to recover, how to protect against future occurrences, and what tools will be used to assist with these activities.
- Define policies for protecting sensitive information and for sharing information with various parties—other sites involved in the incident, other incident response teams, and so on.
- Identify points of contact, both inside and outside your organization. Ensure that those responsible for system and network administration and for incident response know whom to call and under what conditions.
- Define who has hands-on responsibility for dealing with system administration and auditing, and incident response. Define their limits of authority and guidelines for acting in an emergency without consulting a superior (for example, under what conditions should they disconnect from the network?) Write guidelines for related topics, such as who has authority to speak with the press or work with law enforcement.

- Create guidelines for setting priorities among incidents and setting priorities on action items connected to a single incident.
- Determine the conditions for "closing" an incident. When do you stop working on it?

Develop an ongoing relationship with your upstream ISP(s)

If your organization experiences a DoS attack, your upstream Internet service provider(s) may be in a far more advantageous position to mitigate the attack than you are. However, if the first time you've ever called their help desk is during an attack, you might not be satisfied with their response. Thus, it is best to establish a relationship with them in advance so that you know what to do and whom to contact when the time comes.

Provider-customer relationships are built on the service contract. If possible, try to negotiate your ISP's response to a DoS attack on your network as part of the contractual obligation. Here are some items you might want to ask for:

- Visibility of their backbone performance data – If your Internet access is slow, is it because of an attack on your systems or an attack upstream that is saturating your ISP's network?
- Relief from per-bit rate pricing in the event of a DoS attack – It's a long shot, but if you are paying for tiered throughput and you're the target of a DoS attack, your service charges could skyrocket.
- Rate limiting – Either permanently or as-needed, rate limiting (also called. traffic shaping) can help diffuse the effects of a DoS attack.
- Protocol or port blocking – If there are entire protocols or ports that you know you don't need, your ISP may be willing to block them before they even get to you. You might need to request this as a temporary measure during an attack if, for instance, you are the victim of an ICMP (ping) flood and need to shut off ICMP until the attack subsides.
- Response-time commitments for support – It may be possible to negotiate, for a premium, maximum response times for support calls. This is especially important during a debilitating DoS attack in order to restore service as soon as possible.

Beyond the contractual obligations between you and your provider, it is a good idea to identify (and document in your incident response plan) key contacts at the ISP in case of emergency. Knowing who to call, along with what information you'll need to provide, can greatly reduce the time required to mitigate an attack.

4. React

The exact procedure to follow in the event your network suffers a DoS attack will depend on many factors: your network architecture, the nature of the attack traffic, the tools available to diagnose the attack, the filtering abilities of your chosen hardware, the skills and talents of your network staff, the business purpose of the target and the amount of collateral damage being generated, whether your upstream link is saturated or not, your policies, and the advice of your lawyers, not to mention the impact of the attack on your customers, your public image, and your bottom line. At best you will have considered these issues well ahead of time and will have a battle plan in place to guide your actions. While it is not possible to offer a generic recipe for recovering from every type of attack, it is hoped that this section will help you consider your options—both in advance and when "under the gun"—from a more informed perspective and craft an optimal response plan that makes sense in your circumstances.

Generally, for modest levels of attack traffic, you may find it within your means to mitigate the attack yourself. Firewall rules, router ACLs, rate limiting, black hole routing at your network borders and perhaps changing your network topology to sidestep the attack are the techniques you will typically use to minimize the impact of the attack traffic. It is rarely the case however that these methods will completely restore network performance to pre-attack levels, since the attack traffic will still be impinging on your border and consuming your finite upstream bandwidth. But keeping the attack traffic out of your networks and away from the target hosts will allow your regular operations to continue relatively unhindered. Once you have regained the upper hand on internal functionality, or in parallel with this effort, you can notify your upstream provider and see if they can offer any further relief from the attack traffic.

For large attacks that saturate your Internet links, the assistance of your upstream providers will be essential to bring traffic levels down to manageable levels. For still larger attacks your upstreams may require help from their upstreams or peers. Keep in mind that the more assistance you require, the longer it is likely to take to coordinate an effective response. This is where the pre-planning mentioned in earlier sections of this paper will prove its value. Knowing in advance whom to contact within your ISP's organization, what steps they can take on your behalf, and what information they will require from you to assist you, all serves to optimize your organization's response to the attack and minimize the pain associated with the attack.

Note that, while ideally your ISP will be able to trace the flood traffic back to its point(s) of origin and surgically filter it from your Internet feed, in practice the traceback process is far from perfect and filtering is often far from surgical. Unless the attack traffic is originating solely within your ISP's own netblocks, they will need to depend on the cooperation of their upstream or peer providers to trace attack flows back to the next hop, and so on back to the source. Unfortunately the infrastructure within the ISP community for identifying the ingress points of flood traffic into their networks is currently immature, and existing procedures are often cumbersome, labor-intensive and time-consuming. These factors and others, such as language barriers or conflicting business priorities, often stymie the best intentions of service providers and frustrate mitigation efforts. It is often impossible for your provider to trace the attack beyond a certain ingress point in their networks, resulting in filters being applied on your behalf that also filter legitimate traffic. This can impact the subset of your customers whose traffic also reaches you through the filtered networks. Thus there are tradeoffs that must often be considered in order to ride out an attack with as much grace as possible.

In general, successful responses to DoS incidents follow a Detect, Characterize, Mitigate cycle. Detecting an attack, and identifying it as such, triggers a thorough analysis of the attack traffic, hopefully producing information that you or your upstream provider can use to reduce or eliminate the effects of the attack.

Detect

A DoS attack can be detected via normal monitoring of inbound traffic volumes and other performance metrics. However, the first indication of attack often comes from internal help desk calls reporting that one or more services have become unavailable, or from external customers unable to contact your public web server. Upon examination, traffic volumes on the various network segments leading to the attack target may be found to be far higher than normal, perhaps saturated, or the target server's incoming connection queue may be filled, rendering the server unresponsive. Other substantiating evidence may be present, such as a marked increase in dropped packets on some segments or a substantial increase in firewall log entries. External connectivity may suffer, perhaps causing DNS lookups to fail and thus many second order internal failures. A DoS attack is generally not subtle, and makes itself known in ways that are hard to miss.

Note that it is also important to keep an eye on your outbound network utilization numbers. This will help you detect the situation where an intruder has commandeered a compromised machine inside your perimeter and is using it to generate flood traffic against an external host. Although this paper does not focus on floods originated against others from your networks, this possibility is mentioned since the perceived impact on your networks can be very similar to an inbound flood. Your customers won't care whether their inability to contact your web site is caused by excessive inbound or excessive outbound traffic.

Characterize

At this point it is important to attempt to classify the incoming traffic with the goals of confirming the attack and finding some distinguishing characteristic of the attack traffic that you or your upstream providers can use to implement a filter. The methods you use to gain insight into the nature of the attack will depend on the features available in your chosen network hardware. Traffic statistics counters in your network hardware can shed light on the types of packets coming into your networks. For example, using your network management software to sample your border router's SNMP variable showing the number of inbound ICMP packets from your upstream link each minute can tell you when this traffic has increased by 3 orders of magnitude, usually a good indicator that an attack is underway. Other variables track other types of packets, and polling these in a similar manner can provide useful insights into what is happening on your wires.

If your hardware supports it, a better option is to examine recent network flow information. A "flow" in this context is simply a sequence of packets sent from one network endpoint to another. For example, a terminal session between a client at 10.0.0.1 and a telnet server at 10.0.0.2 would consist of two flows - one describing the packets sent from the client endpoint 10.0.0.1:1025/TCP to the server endpoint 10.0.0.2:23/TCP, and another describing the packets sent from the telnet server back to the client. The number of transmitted packets is recorded for each flow. This information is then either dumped periodically to a log file or, in the case of some routers, cached internally to aid in expediting routing packet switching decisions. In either case you can inspect this information to get an idea of the types of traffic that have recently traversed your networks.

Besides showing the source and destination addresses, protocols and perhaps ports of all inbound and outbound traffic, flows can be mined for detailed information essential to characterizing an attack. For example, during a SYN flood against your public web server you might see two flows that look like this:

SRC		**DST**	**Protocol**	**# Pkts**
172.16.26.7:1028	->	10.0.0.2:80	TCP	828335
10.0.0.2:80	->	172.16.26.7:1028	TCP	62983

Since the SRC and DST addresses are mirrors of each other and the port associated with your internal address 10.0.0.2 is 80, these flows are likely to represent an HTTP conversation between the two addresses. Note that the number of packets sent from the browser to the server (80/TCP) is many times greater than the number of packets being returned to the browser. This imbalance is unusual for web traffic, and is a signature of a common SYN flood, wherein the web server's incoming connection queue has become saturated with outstanding connection requests (uncompleted 3-way handshakes) and the server is no longer responding to each incoming SYN packet with a SYN-ACK packet—the heavier the attack, the greater the imbalance. Note that the constant browser address does not necessarily identify the actual attacking machine, nor does it even indicate that only a single machine is generating the SYNs, as source addresses can be and often are forged so as to hide the true source of the attack. Nonetheless this information provides us with the lever we need to craft a filter that will block or shunt all traffic from 172.16.26.7, preventing the SYNs from reaching their intended target and permitting the server to be responsive once again.

More complicated flow lists would result from other types of attacks. A distributed UDP attack designed to saturate your network links might use random source addresses, random source ports, random destination ports, and perhaps even random destination addresses constrained to fall within the target netblock. This flood technique makes it very difficult for the victim to construct a filter that blocks only the attack traffic. Such traffic also produces an enormous flow list and may drive up CPU and memory utilization on your border router as it tries to keep track of all the flows. In this case, studying the flow records may not reveal a surgical method for filtering the attack traffic, and more brute-force methods may be required.

Examining flow records may also reveal other indicators that should arouse suspicion. For example, if IRC is not used or is prohibited by policy in your organization, yet you notice outbound flows from internal hosts to external IRC servers (usually at 6667/TCP), this may be an indication that you have compromised machines on your network. Recent trojan backdoor programs being installed by intruders often connect to an IRC server and are controlled via IRC commands. These programs often include denial-of-service functions that can be remotely initiated by the intruder, and so are of particular note when discovered.

Your toolkit should also include a method for capturing live traffic samples on key network segments and presenting the various packet fields in a manner that allows recognition of unusual or suspicious patterns. Such packet analysis can aid in the identification of common characteristics present in flood packets, such as anomalous combinations of packet header flags or obviously incorrect checksum fields. (e.g., all zero.) Identifying even a single unusual characteristic present only in the flood packets provides the lever required to construct an effective firewall rule or router ACL that will eliminate the unwanted traffic.

Mitigate

Once you understand the nature of the attack, mitigation can be attempted by following the decision tree below. If at any time in this process you reach a point where performance has become acceptable to you, the task is complete. It is important to recognize however that instant mitigation is not always possible. One potential result of your analysis may be the identification of weak points in your present architecture. If so, any successful mitigation strategy will require the upgrade, replacement, or redesign of the weak points or architecture.

An important point to consider at this time is whether it is useful to attempt to trace the attack traffic back to its sources. There are several reasons why this might be desired—legal recourse, determining optimal filtering location, or notifying owners of the compromised machines that are attacking you. Your service providers must be notified of this requirement while the attack is running to allow tracing to be conducted.

You must now determine whether all your upstream links are being saturated by the attack traffic. If so, your only recourse is to contact your upstream providers for assistance. No locally-implemented mitigation strategies can be effective until this prerequisite has been met. If a subset of your upstream links are not saturated, routing policy changes may be effective in redistributing the load.

If you request help from your service providers, you must provide them with the attack and flow data that you captured earlier. Other information such as the estimated start time of the attack, the intensity of the attack, and the damage caused by the attack will also be helpful.

You should next determine if the attack traffic is filterable based on a pattern present in each flood packet. If you answer NO to this question, your service providers will also be unlikely to be able to filter the attack traffic. In this case more extreme measures may be required, such as readdressing the target or blocking all traffic to the target. If however you answer YES, you should attempt to implement your filtering strategy.

Before you commit to a filtering solution, realize that there are many subtle engineering issues that must be considered. For example, does your firewall have sufficient CPU power and interrupt handling capacity to filter the full flood traffic, or does the incoming traffic need to first be rate-limited to avoid swamping the firewall? Any filtering solution should also include a back-out plan. This should be used in the event that the filtering solution fails to mitigate the attack or introduces additional problems.

The method for implementing your filter will depend on the device selected to perform the filtering. This is an engineering decision that will be unique to each network architecture and attack type. Assuming the filtering device is in place, implement the filtering solution and immediately test the effect. Be prepared to quickly enact the back-out plan if negative results are observed.

Measuring the success of the filtering solution must be performed at regular intervals for the duration of the attack. This will detect changes in the attack strength or type that may render the filtering solution ineffective. These measurements will also help to detect when the attack ends. Measuring the effect of the filtering solution on the filtering device must also be performed at regular intervals for the duration of the attack. This will detect any performance degradation of the filtering device and alert you that additional measures may be required.

If the filtering solution was ineffective or caused additional problems, the cause must be determined. This may require additional analysis of the attack and its effect on your equipment. It is also possible that the capabilities of the filtering devices were misunderstood or that their performance envelope was exceeded. From this additional analysis effort a new filtering solution may be attempted, or the process may be aborted if another filtering solution can not be found.

If the filtering solution was successful, monitoring and forensic data collection should be continued for the duration of the attack to aid in post-mortem analysis. Termination of observed attack traffic does not necessarily indicate that the attacker has given up. Continued monitoring is advised in order to detect resumption of the attack.

A post-mortem analysis should always be conducted. The goals should include

- determining the attack type, source, and likely cause.
- determining the attack's effect on the intended target as well as collateral damage to other resources.
- gauging organizational reaction - were appropriate people and resources allocated to the problem?
- determining if documented processes were followed. Were they effective?
- identifying how the attack was detected. Can detection be improved?
- assessing the damages and determining the cost of the attack.
- determining legal recourse, if any.
- assessing the responsiveness of external parties during the event.

In summary, not all attacks against a given network architecture have an instant solution. However following the above process will either yield effective mitigation techniques or uncover weak points in the architecture. Addressing these weak points will improve responsiveness to future incidents.

5. Possibilities for the Future

There are a number possibilities for the future that might provide some relief from denial-of-service attacks. In this section, we consider commercial activities, research, and protocol development.

Commercial Developments

Commercial products are available today to help with detecting and reacting to DoS attacks. In general, these products monitor network traffic for attack signatures and/or anomalous traffic that may indicate a DoS attack in progress. These products then may alert administrators and recommend, or even perform, configuration changes such as rate-limiting filtering. It is important to note that these products only work within the network where they are deployed, so while they might alleviate traffic internally, they have no effect on traffic coming from an ISP and cannot filter or limit an attack at that point. A more promising approach is for an ISP to employ such technology, which might make that ISP more attractive to concerned customers.

Good netizenship today dictates that customers perform egress filtering to prevent traffic with improper source addresses from leaving their networks. Since legitimate traffic from your network will always have source addresses from your assigned address space, traffic with spoofed source addresses should not be allowed to leave your network. Egress filtering at your network border can prevent traffic with spoofed source addresses from reaching the Internet and ensures that traffic from your network can be traced back to its true point of origin. This behavior could become more widespread in the future if it were enforced, possibly as part of the agreements between the network and its ISPs.

Other changes in filtering activities are suggested by ISP behavior during recent outbreaks of the Code Red and Nimda worms, when the ISPs disconnected customers who were infected. Network operators discuss “blackholing” entire networks, and some mail administrators do not accept connections from blacklisted servers. A similar approach could be applied to egress filtering, in which sites are quarantined if they pass on spoofed traffic.

Promising Research

While backscatter analysis does nothing to stop or mitigate any one DoS attack, research in this area has helped to quantify the frequency and scope of DoS attacks. Backscatter analysis is based on a set of assumptions, one of which is that attacks use random spoofed source addresses. Therefore, backscatter analysis does not count attacks that do not spoof source addresses. It is important to take different creative approaches in analyzing DoS attacks. While many attacks today may be characterized at the packet level, better attack tools may generate traffic that is not as easily identified.

Protocol Developments

Some of the following activities may be fruitful directions for the future. They are worth watching. We have provided URLs wherever possible.

- IPSEC
  IPSEC can be used to provide authentication, integrity, and confidentiality for IP traffic. To be useful, IPSEC must be deployed at both ends of a connection.

  Authentication Header (AH): http://www.ietf.org/rfc/rfc2402.txt
  Encapsulating Security Payload (ESP): http://www.ietf.org/rfc/rfc2406.txt

- ICMP Traceback (itrace)
  ICMP Traceback, or itrace, is a draft protocol designed to aid in tracking down attack agents that spoof their source addresses. Using traceback, routers generate relatively infrequent ICMP traceback messages that identify traffic flows. By analyzing traceback messages from many routers, the target of an attack can better identify the source(s) of the attack. For further information, see the itrace Internet draft:

  http://www.ietf.org/internet-drafts/draft-ietf-itrace-00.txt

  See also:

  Practical Network Support for IP Traceback by Stefan Savage, David Wetherall, Anna Karlin, Tom Anderson
  http://www.cs.washington.edu/homes/savage/papers/Sigcomm00.pdf

  Advanced and Authenticated Marking Schemes for IP Traceback by Adrian Perrig and Dawn Song
  http://paris.cs.berkeley.edu/~perrig/projects/iptraceback/tr-iptrace.pdf

- Pushback
  Pushback is a protocol that enables downstream routers to notice a possible DoS attack and alert upstream routers, asking them to limit or drop attack traffic.

  Pushback Messages for Controlling Aggregates in the Network
  Internet Engineering Task Force
  http://www.ietf.org/internet-drafts/draft-floyd-pushback-messages-00.txt
  http://www.aciri.org/pushback/

- Host Identity Payload and Protocol (HIP)
  HIP is a protocol that provides authentication for IP traffic. HIP is designed to work with ESP in conjunction with IPSEC. HIP would be useful in mitigating TCP SYN flood attacks, since it lowers the cost of the target's response to an initial SYN packet.

  Host Identity Payload and Protocol
  Internet Engineering Task Force
  Robert Moskowitz, TrueSecure Corporation
  http://search.ietf.org/internet-drafts/draft-moskowitz-hip-04.txt

- IPv6
  Deployment of the next version of Internet Protocol (IP) will certainly change the behavior of the Internet, including DoS attacks and defenses. IPSEC features and congestion control (ECN) will be readily available. Exponentially increased address space will decrease the effectiveness of attackers scanning for vulnerable hosts to claim as DoS agents. Allocation of addresses directly to end-users will help identify attacking sites, unlike today's allocations to ISPs. Flow labeling (RFC 2460) could be used to identify aggregate traffic flows, such as those caused by DoS attacks. The IPNG Working Group is discussing if and how to implement traceback features. Another idea under discussion is ingress filtering at the customer edge of ISPs. By enforcing topological correctness at the first hop, attackers will be unable to spoof their source addresses.

  IPNG Working Group
  Internet Engineering Task Force
  http://www.ietf.org/html.charters/ipngwg-charter.html

- Border Gateway Protocol (BGP)
  BGP is the most common routing protocol used between multi-homed networks. BGP makes decisions based on link-state and distance-vector calculations. A new protocol that is aware of available bandwidth could help keep a target site connected by routing legitimate traffic to less utilized links. New versions of this protocol could be promising if they increase their ability to sidestep heavy network traffic.

6. Conclusion

The most fundamental lesson to be learned from distributed denial of service is the fact that all sites on the Internet are interdependent, whether they know it or not. The impact upon your site and its operations is dictated by the (in)security of other sites and the ability of a remote attacker to implant the tools and, subsequently, to control and direct multiple systems worldwide to launch an attack.

Attackers typically exploit well-known vulnerabilities, many of which have readily available fixes. Complicating matters are the intrusion tools that are widely available. Intruders have automated the processes for discovering vulnerable sites, compromising them, installing daemons, and concealing the intrusion.

Even security-conscious sites can suffer a denial of service because attackers can control other, more vulnerable computer systems and use them against the more secure site. Thus, although you may be able to "harden" your own systems to help prevent having them used as part of a distributed attack, currently available technology does not enable you to avoid becoming a victim. There is some hope for the future in technological and other approaches.

Handling denial of service is essentially an exercise in risk management. There are sometimes technical solutions to management problems. There are always management solutions to technical problems. We encourage readers to look at denial of service from both points of view.

Appendix: A More Detailed Look at Attacks

DoS attacks exploit the asymmetric nature of certain types of network traffic. One attack method seeks to cause the target to use more resources processing traffic than the attacker does sending the traffic. Another method is to control multiple attackers.

Types of Denial-of-Service Attacks

There are several general categories of DoS attacks. Some groups divide attacks into three classes: bandwidth attacks, protocol attacks, and logic attacks.
Following are brief descriptions of some common types of DoS attacks.

Bandwidth/Throughput Attacks

Bandwidth attacks are relatively straightforward attempts to consume resources, such as network bandwidth or equipment throughput. High-data-volume attacks can consume all available bandwidth between an ISP and your site. The link fills up, and legitimate traffic slows down. Timeouts may occur, causing retransmission, generating even more traffic. An attacker can consume bandwidth by transmitting any traffic at all on your network connection. A basic flood attack might use UDP or ICMP packets to simply consume all available bandwidth. For that matter, an attack could consist of TCP or raw IP packets, as long as the traffic is routed to your network.

A simple bandwidth-consumption attack can exploit the throughput limits of servers or network equipment by focusing on high packet rates—sending large numbers of small packets. High-packet-rate attacks typically overwhelm network equipment before the traffic reaches the limit of available bandwidth. Routers, servers, and firewalls all have constraints on input-output processing, interrupt processing, CPU, and memory resources. Network equipment that reads packet headers to properly route traffic becomes stressed handling the high packet rate (pps), not the volume of the data (Mbps). In practice, denial of service is often accomplished by high packet rates, not by sheer traffic volume.

Protocol Attacks

The basic flood attack can be further refined to take advantage of the inherent design of common network protocols. These attacks do not directly exploit weaknesses in TCP/IP stacks or network applications but, instead, use the expected behavior of protocols such as TCP, UDP, and ICMP to the attacker's advantage. Examples of protocol attacks include the following:

- *SYN flood* is an asymmetric resource starvation attack in which the attacker floods the victim with TCP SYN packets and the victim allocates resources to accept perceived incoming connections. As mentioned above, the proposed Host Identity Payload and Protocol (HIP) is designed to mitigate the effects of a SYN flood attack. Another technique, SYN Cookies (see http://cr.yp.to/syncookies.html), is implemented in some TCP/IP stacks.

- *smurf* is an asymmetric reflector attack that targets a vulnerable network broadcast address with ICMP ECHO REQUEST packets and spoofs the source of the victim (see http://www.cert.org/advisories/CA-1998-01.html).
- *fraggle* is a variant of smurf that sends UDP packets to echo or chargen ports on broadcast addresses and spoofs the source of the victim.

Software Vulnerability Attacks

Unlike flooding and protocol attacks, which seek to consume network or state resources, logic attacks exploit vulnerabilities in network software, such as a web server, or the underlying TCP/IP stack. Some vulnerabilities by crafting even a single malformed packet.

- *teardrop (bonk, boink)* exploits TCP/IP IP stacks that do not properly handle overlapping IP fragments (see http://www.cert.org/advisories/CA-1997-28.html).
- *land* crafts IP packets with the source address and port set to be the same as the destination address and port (see http://www.cert.org/advisories/CA-1997-28.html).
- *ping of death* sends a single large ICMP ECHO REQUEST packet to the target.
- *Naptha* is a resource-starvation attack that exploits vulnerable TCP/IP stacks using crafted TCP packets. (see http://www.cert.org/advisories/CA-2000-21.html).

There are many variations on these common types of attacks, and many varieties of attack tools to implement them.

You can find more information about vulnerabilities in the CERT/CC Vulnerability Notes Database, which is available to the public (http://www.kb.cert.org/vuls/) and in the Vulnerability Catalog, which is available to authorized users (https://www.kb.cert.org/vulcatalog/).

Denial-of-service attacks may be effective because of a combination of effects. For example, an attack that does not fully consume bandwidth or overload equipment throughput may be effective because it generates enough malformed traffic to crash a particular service, such as a web server or mail server (for an example, see http://www.cert.org/advisories/CA-2000-21.html).

Root Causes of Attacks

Many attacks consist of large numbers of hosts, or computers, operating under the control of the attacker. These hosts may be referred to as zombies, agents, slaves, or bots. The huge number of hosts connected to the Internet gives attackers plenty of potential attack agents that are vulnerable to compromise. Root causes include the level of security at individual sites, the nature of attack tools, and vulnerabilities in software products.

Level of security

Sites cannot be 100 percent safe if it they connect to the Internet. Secure and well-managed sites are not likely to be compromised and if they are, the compromise is not likely to remain undetected for long. The systems at an ideal site like this are up to date on patches, have a firewall, are monitored, have unneeded services and features disabled, and have antivirus software installed, configured, and up to date. The ideal site also has an incident response capability and skilled staff. Being a good netizen, the site does not permit spoofed traffic to leave its network.

Unfortunately, most systems fall short of the ideal. Systems at educational institutions, for example, frequently have lots of available bandwidth but might have less administrative oversight, especially in the case of student-owned computers. Similarly, a typical home user with a broadband connection most likely is not aware of the security implications of being connected to the public network. In the workplace, some administrators are more security-minded than others and some supervisors place a higher priority on security than others. The manner and degree in which a machine is managed directly contributes to that machine's potential for abuse as a DoS agent.

Attack tools

Attack tools are controlled in a variety of ways. Some earlier tools established listening ports (trinoo). A more subtle control channel uses ICMP echo reply packets (Tribe Flood Network, TFN). Recent tools exploit the existing infrastructure of Internet Relay Chat (IRC) networks. These tools are not as easily discovered as earlier ones, as they connect to an IRC channel and do not present an open port that could be found by a scan or audit.

Attack tools can also be installed through Trojan horse programs that users have been convinced to download and execute (Leaves, Knight). Moreover, exploits (programs or scripts used for exploiting vulnerabilities at a site) are packaged in readily available formats that allow even neophyte attackers to create networks of attack agents.

Software vulnerabilities

Many hosts used as DoS attack agents are subverted through well-known vulnerabilities in commonly used software (as seen in incidents involving the Code Red worm and Power malicious code). Insecure design of common web browsers and email software exacerbate the problem. Both types of problems are noted in CERT advisory CA-2001-20 (http://www.cert.org/advisories/CA-2001-20.html).

A significant number of these vulnerabilities allow attackers to gain root/administrator access from a remote location. Attackers gain complete control over the computer and may use it to suit their purpose, which may be to use it as a DDoS agent. Current economic pressures lead vendors to focus on achieving a fast time to market rather than on designing secure networks and applications. Without some financial (or legal) incentive to behave more securely, developers will continue to produce vulnerable products.

Anonymity

It is easy for attackers to avoid getting caught by hiding their identity. They command their attack network from stolen dial-up accounts and other compromised systems, and they use spoofed source addresses for attack traffic. Victim sites and law enforcement face a daunting and frequently unfeasible task to identify and prosecute attackers. Suffering few consequences—if any—for their actions, attackers continue their work.

The combination of all of these factors provide a fertile environment for DoS agents.

References and Additional Reading

CERT® Coordination Center References

An Analysis of Security Incidents on the Internet 1989-1995
Chapter 11, "Denial-of-Service Incidents"
Dr. John D. Howard
http://www.cert.org/research/JHThesis/Chapter11.html

CERT advisory CA-1996-01: UDP Port Denial-of-Service Attack
http://www.cert.org/advisories/CA-1996-01.html

CERT advisory CA-1996-21: TCP SYN Flooding and IP Spoofing Attacks
http://www.cert.org/advisories/CA-1996-21.html

CERT advisory CA-1996-26: Denial-of-Service Attack via ping
http://www.cert.org/advisories/CA-1996-26.html

CERT advisory CA-1997-28: IP Denial-of-Service Attacks
http://www.cert.org/advisories/CA-1997-28.html

CERT advisory CA-1998-01: Smurf IP Denial-of-Service Attacks
http://www.cert.org/advisories/CA-1998-01.html

CERT advisory CA-1998-13: Vulnerability in Certain TCP/IP Implementations
http://www.cert.org/advisories/CA-1998-13.html

CERT advisory CA-1999-04: Melissa Macro Virus
http://www.cert.org/advisories/CA-1999-04.html

CERT advisory CA-1999-17: Denial-of-Service Tools
http://www.cert.org/advisories/CA-1999-17.html

CERT advisory CA-2000-01: Denial-of-Service Developments
http://www.cert.org/advisories/CA-2000-01.html

CERT advisory CA-2000-11: MIT Kerberos Vulnerable to Denial-of-Service Attacks
http://www.cert.org/advisories/CA-2000-11.html

CERT advisory CA-2000-20: Multiple Denial-of-Service Problems in ISC BIND
http://www.cert.org/advisories/CA-2000-20.html

CERT advisory CA-2000-21: Denial-of-Service Vulnerabilities in TCP/IP Stacks
http://www.cert.org/advisories/CA-2000-21.html

CERT advisory CA-2001-18: Multiple Vulnerabilities in Several Implementations of the Lightweight Directory Access Protocol (LDAP)
http://www.cert.org/advisories/CA-2001-18.html

CERT advisory CA-2001-20: Continuing Threats to Home Users
http://www.cert.org/advisories/CA-2001-20.html

CERT incident note IN-99-07: Distributed Denial of Service Tools

http://www.cert.org/incident_notes/IN-99-07.html
CERT incident note IN-2000-01: Windows Based DDOS Agents
http://www.cert.org/incident_notes/IN-2000-01.html

CERT incident Note IN-2000-02: Exploitation of Unprotected Windows Networking Shares
http://www.cert.org/incident_notes/IN-2000-02.html

CERT incident note IN-2000-04: Denial of Service Attacks using Nameservers
http://www.cert.org/incident_notes/IN-2000-04.html

CERT incident note IN-2000-05: "mstream" Denial of Service Tool
http:// www.cert.org/incident_notes/IN-2000-05.html

CERT incident note IN-2000-10: Widespread Exploitation of rpc.statd and wu-ftpd Vulnerabilities
http://www.cert.org/incident_notes/IN-2000-10.html

CERT incident note IN-2001-04: "Carko" Distributed Denial-of-Service Tool
http:// www.cert.org/incident_notes/IN-2001-04.html

CERT security improvement module SIM-009: Detecting Signs of Intrusions
http://www.cert.org/security-improvement/

CERT security improvement module SIM-006: Responding to Intrusions
http://www.cert.org/security-improvement/

*The practices documented in the above two modules have been published in **The CERT Guide to System and Network Security Practices** by Julia Allen, Addison-Wesley, 2001.*

Denial of Service Attacks (tech tip)
http://www.cert.org/tech_tips/denial_of_service.html

Results of the Distributed-Systems Intruder Tools Workshop
CERT Coordination Center and others
http://www.cert.org/reports/dsit_workshop-final.html
http://www.cert.org/reports/dsit_workshop.pdf

Trends in Denial of Service Attack Technology
CERT Coordination Center in collaboration with Neil Long and Rob Thomas
http://www.cert.org/archive/pdf/DoS_trends.pdf

Vulnerability Catalog
https://www.kb.cert.org/vulcatalog/
(Authorized access only. ISA members with valid certificates are authorized.)

Vulnerability Notes Database
http://www.kb.cert.org/vuls/

Other References

Host Identity Payload and Protocol
Internet Engineering Task Force
Robert Moskowitz, TrueSecure Corporation
http://search.ietf.org/internet-drafts/draft-moskowitz-hip-04.txt

ICMP Traceback Working Group
Internet Engineering Task Force
http://www.ietf.org/html.charters/itrace-charter.html
http://www.ietf.org/internet-drafts/draft-ietf-itrace-00.txt

IPNG Working Group
Internet Engineering Task Force
http://www.ietf.org/html.charters/ipngwg-charter.html

Inferring Internet Denial-of-Service Activity
CAIDA, University of California, San Diego
http://www.caida.org/outreach/papers/backscatter/

The Naptha DoS Vulnerabilities
BindView RAZOR
http://razor.bindview.com/publish/advisories/adv_NAPTHA.html
http://razor.bindview.com/publish/advisories/adv_list_NAPTHA.html

Pushback Messages for Controlling Aggregates in the Network
Internet Engineering Task Force
http://www.ietf.org/internet-drafts/draft-floyd-pushback-messages-00.txt
http://www.aciri.org/pushback/

Additional Reading

Advanced and Authenticated Marking Schemes for IP Traceback
Adrian Perrig, Dawn Song
http://paris.cs.berkeley.edu/~perrig/projects/iptraceback/tr-iptrace.pdf

CenterTrack: An IP Overlay Network for Tracking DoS Floods
Robert Stone, UUNet
http://www.nanog.org/mtg-9910/robert.html

Cisco IOS NetFlow
Cisco
http://www.cisco.com/warp/public/732/Tech/netflow/

Countering DCAs
Fred Cohen
http://all.net/journal/netsec/2000-04.html

DDoS – Is There Really a Threat?
Dave Dittrich
http://staff.washington.edu/dittrich/talks/sec2000/

Distributed Denial of Service (DDoS) Attacks/Tools
http://staff.washington.edu/dittrich/misc/ddos/

Distributed Denial of Service Defense Tactics
Simple Nomad, BindView RAZOR
http://razor.bindview.com/publish/papers/DDSA_Defense.html

DoS Links
http://www.denialinfo.com/

A Framework for Denial of Service Analysis
Catherine Meadows, Naval Research Laboratory
http://www.cert.org/research/isw/isw2000/papers/37.pdf

"Hot Spares" For DoS Attacks
Tina Darmohray and Ross Oliver
http://www.usenix.org/publications/login/2000-7/apropos.html

Monitoring DoS Attacks with the VIP Console and NetFlow v1.0
Rob Thomas
http://www.cymru.com/~robt/Docs/Articles/dos-and-vip.html

Practical Network Support for IP Traceback
Stefan Savage, David Wetherall, Anna Karlin, Tom Anderson
http://www.cs.washington.edu/homes/savage/papers/Sigcomm00.pdf

Strategies for Defeating Distributed Attacks
Simple Nomad, BindView RAZOR
http://razor.bindview.com/publish/papers/strategies.html

Tracking Spoofed IP Addresses Version 2.0
Rob Thomas
http://www.cymru.com/~robt/Docs/Articles/tracking-spoofed.html

The XenoService – A Distributed Defeat for Distributed Denial of Service
Jianxin Yan, Stephen Early, Ross Anderson
http://www.cert.org/research/isw/isw2000/papers/42.pdf

all URLs were current as of October 26, 2001