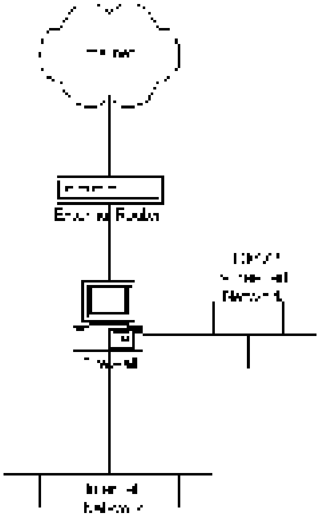
Egress Filtering

Simply stated, egress filtering is the filtering of outbound traffic. Figure 1 shows a typical network with an internal leg and an external screened/DMZ leg. In this scenario it is typical of most administrators to tightly control what is coming to you inbound. However, outbound traffic is often overlooked.



Why is it important not to overlook this? Without proper configuration and consideration of egress filters, your network could unwittingly be used for an attack.

For example: You have done a nice job of locking down your firewall and have limited what traffic can reach your screened network. However, you have not, for one reason or another, applied all of the patches to your server. An intruder launches an attack and runs a known exploit that has not been patched. You system has now been compromised and since you have no limits on what can leave your network the attacker now has a prime host in which to launch further attacks. One of which would be to launch an attack using spoofed IP addresses.

To protect yourself from unwittingly being used in an attack against other systems (and being a good internet neighbor) is to prevent spoofing of your outbound traffic. Let's say that your screened network is using an address space of 198.198.198.0. You would want to block any outbound traffic that did not originate from this network. To do this on a Cisco router you would create an extended access-list containing your address space and apply it inbound to the interface that is facing your network. For example:

```
access-list 101 permit ip 198.198.198.0 0.0.0.255 any
interface eth0
ip access-group 101 in
```

Spoofing attacks that come from your network, be it screened network or internal, have now been denied. This type of filter does not prevent your systems from being hacked, but it does prevent someone from using spoofed IP addresses from your network.

Why is this important? Most DoS attacks use some sort of spoofing when they are run to try and hide their source. In denying this ability to spoof, we have thus denied the ability to use our hosts in this type of attack. If all administrators helped to do this we could make it harder for the would-be attacker.

For more information on egress filtering there is a comprehensive guide written by Chris Brenton located at http://www.sans.org/y2k/egress.htm.

For specific information on configuring your CheckPoint FW-1 firewall for anti-spoofing, refer to http://www.phoneboy.com/fw1/faq/0140.html.

Firewall Policy Violations

The following entries are taken from our firewall log. I have changed some of the information using the following template: Source = S.S.S.x Destination = D.D.D.x Firewall = FW.FW.FW.FW. The last rule of the firewall (rule 27) states drops any traffic that did not pass one of the previous rules. So basically, any source, any destination, any protocol will be dropped.

Firewall Log Entry #1

Date/Time	Interface	Action	Prot.	Service	Source	Destination	Rule
1Jun2000:15:56:01	CpqNF32	Reject	TCP	Telnet	S.S.S.156	D.D.D.67	27

Rule that caught the violation:

Rule #	Source	Destination	Protocol	Action
27	Any	Any	Any	Reject

This violation is an attempt by an outside host to telnet to a host behind the firewall. The written policy states that Telnet is not allowed from any external interface. The firewall is enforcing this policy by rejecting this Telnet attempt.

Potential Damage:

Telnet is a virtual terminal protocol that is used for remote terminal connection service. It allows a user at one site to interact with a system at another site as if that users terminal were directly connected to the host computer itself. The problem arises however in that Telnet sends its information in the clear. It is easily sniffed and hijacked. Since the administration staff resides internally, a best practice is to deny telnet from the external interface.

Firewall Log Entry #2

Date/Time	Interface	Action	Prot.	Service	Source	Destination	Rule
2Jun2000:20:03:32	CpqNF32	Drop	ICMP		S.S.S.33	D.D.D.1	27
2Jun2000:20:03:45	CpqNF32	Drop	ICMP		S.S.S.33	D.D.D.2	27
2Jun2000:20:03:47	CpqNF32	Drop	ICMP		S.S.S.33	D.D.D.3	27
2Jun2000:20:03:52	CpqNF32	Drop	ICMP		S.S.S.33	D.D.D.4	27

Rule that caught the violation:

Rule #	Source	Destination	Protocol	Action
27	Any	Any	Any	Reject

This violation is an attempt by an outside host to gather information about possible target hosts behind the firewall using ICMP/ping. The written policy states that ICMP is not allowed from any external interface. The firewall is enforcing this policy by rejecting these packets.

Potential Damage:

Ping is often used for troubleshooting network problems. However, ping can also be used for reconnaissance. The pattern seen above is a blatant attempt to find available targets behind the firewall. To help prevent this type of information gathering we are rejecting the packets.

Firewall Log Entry #3

Date/Time	Interface	Action	Prot.	Service	Source	Destination	Rule
2Jun2000:20:05:01	CpqNF32	Reject	TCP	Finger	S.S.S.21	D.D.D.34	27

Rule that caught the violation:

Rule #	Source	Destination	Protocol	Action
27	Any	Any	Any	Reject

This violation is an attempt by an outside host to use the finger utility. The last rule in the firewall policy is to deny anything that is not allowed in a previous rule. The written policy states that Finger is not to be used at all. The firewall is enforcing this policy by rejecting this Finger attempt.

Potential Damage:

Finger is a utility available on UNIX computers and included with many TCP/IP protocol suites (for other operating systems) that provides information about users with accounts on the local computer or a remote computer. It would be used for information gathering and potentially social engineering. Once an attacker knows user information such as their full name, then they could potentially use that to their advantage.

Firewall Log Entry #4

Date/Time	Interface	Action	Prot.	Service	Source	Destination	Rule
15May2000:3:22:32	CpqNF32	Reject	TCP	rlogin	S.S.S.114	D.D.D.44	27

Rule that caught the violation:

Rule #	Source	Destination	Protocol	Action
27	Any	Any	Any	Reject

This violation is an attempt by an outside host to remotely login to a host behind the firewall. The last rule in the firewall policy is to deny anything that is not allowed in a previous rule. The written policy states that r-commands are not allowed. The firewall is enforcing this policy by rejecting this remote logon attempt.

Potential Damage:

Remote commands allow you to create trusts between your systems. While this can make an administrators life easier, it creates potential back doors. If an intruder was able to break one host, they automatically have access to every host that first host has. And from there the fun (for them) begins.

Firewall Log Entry #5

Date/Time	Interface	Action	Prot.	Service	Source	Destination	Rule
1Jun2000:15:56:01	CpqNF32	Reject	TCP	TFTP	S.S.S.156	D.D.D.67	27

Rule that caught the violation:

Rule #	Source	Destination	Protocol	Action
27	Any	Any	Any	Reject

This violation is an attempt by an outside host to TFTP to a host behind the firewall. The last rule in the firewall policy is to deny anything is not allowed in a previous rule. The written policy states that TFTP is not allowed from any external interface. The firewall is enforcing this policy by rejecting this Telnet attempt.

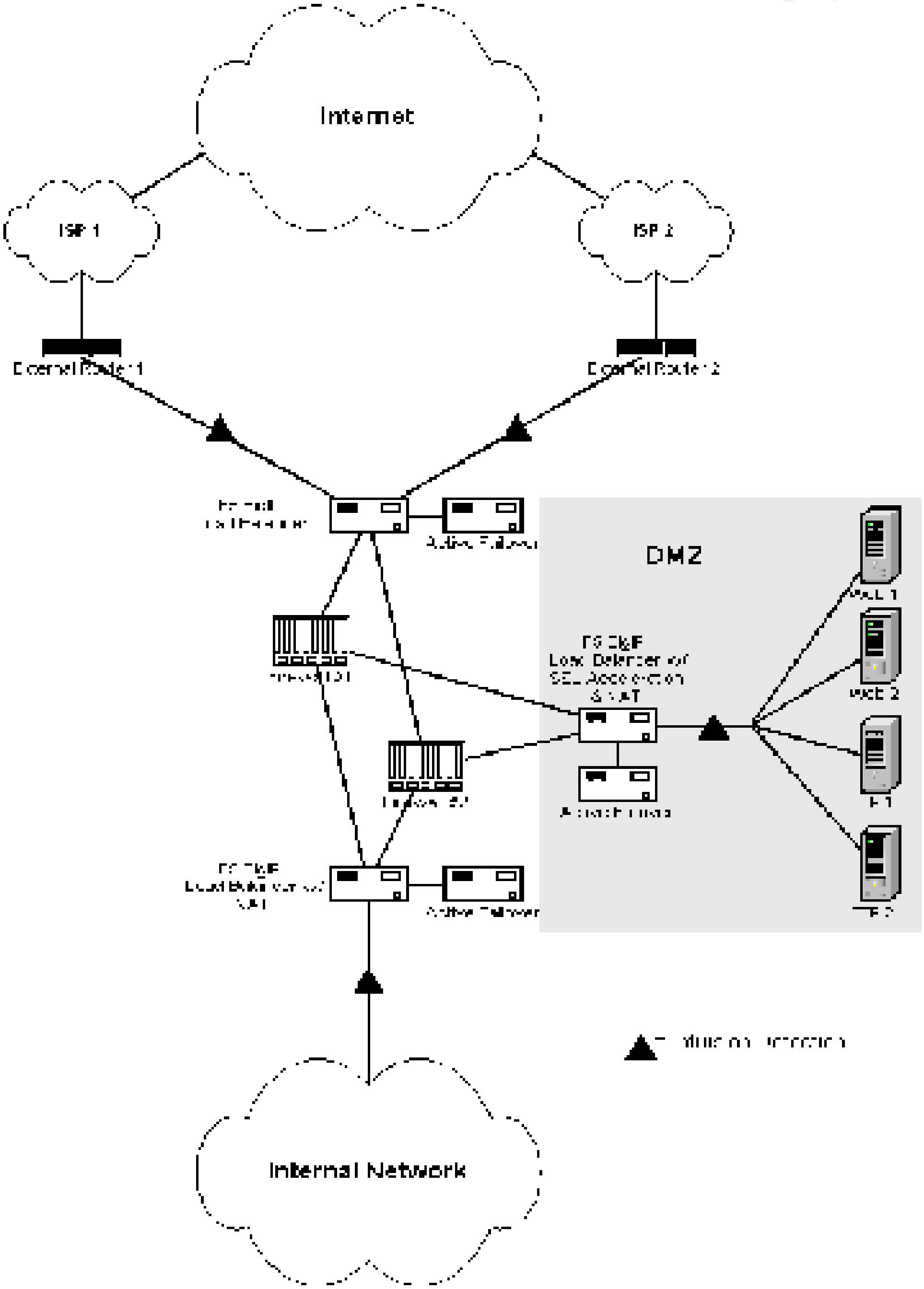
Potential Damage:

"TFTP, or Trivial File Transfer Protocol, is typically used to boot diskless workstations or network devices such as routers. TFTP is a UDP-based protocol that listens on port 69 and provides very little security. Many times attackers will locate a system with TFTP server enabled and attempt to TFTP a copy of the /etc/passwd file back to their file system. If the TFTP server is configured incorrectly, the target system will happily give up the password file. The attackers now have a list of usernames that can be brute forced." -Quote taken from Hacking Exposed

Defense in Depth Architecture

Scenario #1

The following diagram represents a sample network with dual connections to the Internet. In addition, the network has a Screened DMZ that contains multiple Web servers and multiple FTP servers. Upstream ISP's are providing DNS. For simplicity, I have excluded e-mail services.



Overview

In this design, I have tried to eliminate all single points of failure. To do this, I have two different routers connecting to two different ISP's, load balancing between two firewalls, and multiple web/ftp servers running the same application for maximum availability. Since we have gone to the length of getting two ISP's, I have added redundancy throughout.

Perimeter Routers

This network has two Internet connections via two different ISP's. The perimeter routers employ both ingress filters for connecting to specific services on the DMZ and egress filters to prevent spoofing and the use of this site for DDoS attacks. ICMP is not allowed into the network from any external source as part of the ingress filters deny any any rule. Sample access lists are as follows:

Ingress Filters:

```
access-list 101 permit tcp any any 80
access-list 101 permit tcp any any 443
access-list 101 permit tcp any any 21
access-list 101 permit tcp any any 20
access-list 101 deny ip any any log
interface s0
ip access-group 101 out
```

Egress Filters:

```
access-list 104 permit ip 198.198.198.0 0.0.0.255 any
interface eth0
ip access-group 104 in
```

F5 Networks BigIP-HA Load Balancers

In this diagram, there are three sets of load balancers. All three sets are deployed in pairs to prevent them from becoming a single point of failure. One box is active while the other is in active standby mode. Both boxes are keeping track of connections/traffic and during fail-over the second box will keep the state of existing connections. Load balancing is necessary in this case to distribute the traffic between the two firewalls not only incoming from the Internet, but outgoing from the DMZ and Internal Network.

The load balancers that are in front of the DMZ are actually performing multiple functions. First, they are balancing traffic between the web and ftp servers themselves. In this scenario, I have multiple web servers that have the same content and the load is balanced between them. This will prevent the site from being down if one of the web servers should happen to fail or become unavailable. The same holds true for the FTP servers. Secondly, NAT will be performed on the BigIP box so that private, non-routable addresses may be used for the Web and FTP servers. And finally, the BigIP box is doing SSL acceleration to off-load the processor intensive encryption from the Web servers themselves.

The load balancers that are in front of the Internal Network are performing two functions, distribution of outbound traffic and NAT. NAT is performed to give flexibility for addressing the internal network as well as another layer of protection.

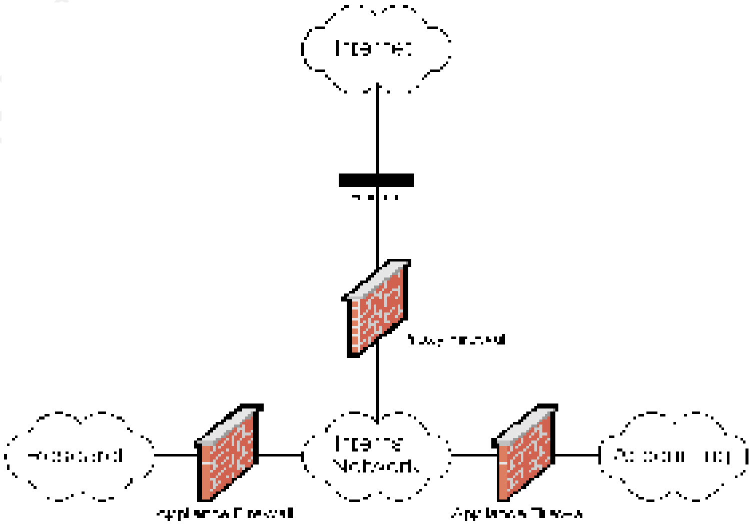
Firewalls
The firewalls have filters to further limit the access by not letting outside traffic to enter into the internal network. The re-iteration of the filters going towards the DMZ are also entered to only allow web and ssl traffic to reach the web servers, and ftp to reach the ftp servers. Those same rules are applied to the internal network, as they should only be connecting to the DMZ via those same ports. The traffic is being distributed between the two firewalls and as the need grows, more firewalls could be added.

Intrusion Detection
Intrusion Detection has been added to further layer our defensive architecture. There are still exploits that can take advantage of poorly configured web servers. To this end, IDS has been put in front of the web/ftp servers to look for patterns of attacks. The placement of this IDS will help us to react to alarms that need immediate action.

Two ID Systems have been placed between the load balancers and the routers for protection as well and to also help us do some trending on what types of attacks are being launched.

The final IDS will be placed on the internal network segment to help prevent our internal users from launching attacks towards the outside and the DMZ. As budgets allow, more Intrusion Detection systems should be placed near or on critical systems. The focus of this drawing was to protect our internal network and our DMZ from external attack.

Scenario #2
This implementation has a few constraints as follows. This site has two critically important internal sub-networks; research and accounting that require a high degree of protection. The site is connected to the Internet. Equipment has already been purchased (Cisco router, proxy firewall, and two appliance firewalls) and cannot be sent back.



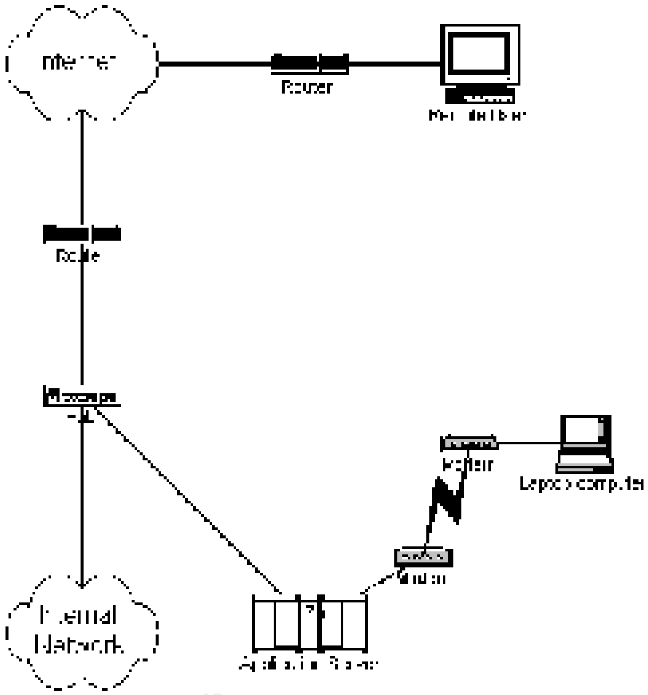
Implementation

With this design, the two "high protection" mandates have been met. Both accounting and research have been put off on their own segments and have the appliance/bridging firewalls placed between them and the internal network. These firewalls will only allow request originating from research or accounting to reach the internal network and Internet. Since they are acting in a bridging fashion, they may still use the proxy server for outbound connectivity. The proxy firewall is helping provide some protection to the internal network for web and ftp traffic. As long as other services are blocked at the router, this can be reasonably secured. Both ingress and egress filters will have to be carefully and properly applied.

Test Questions

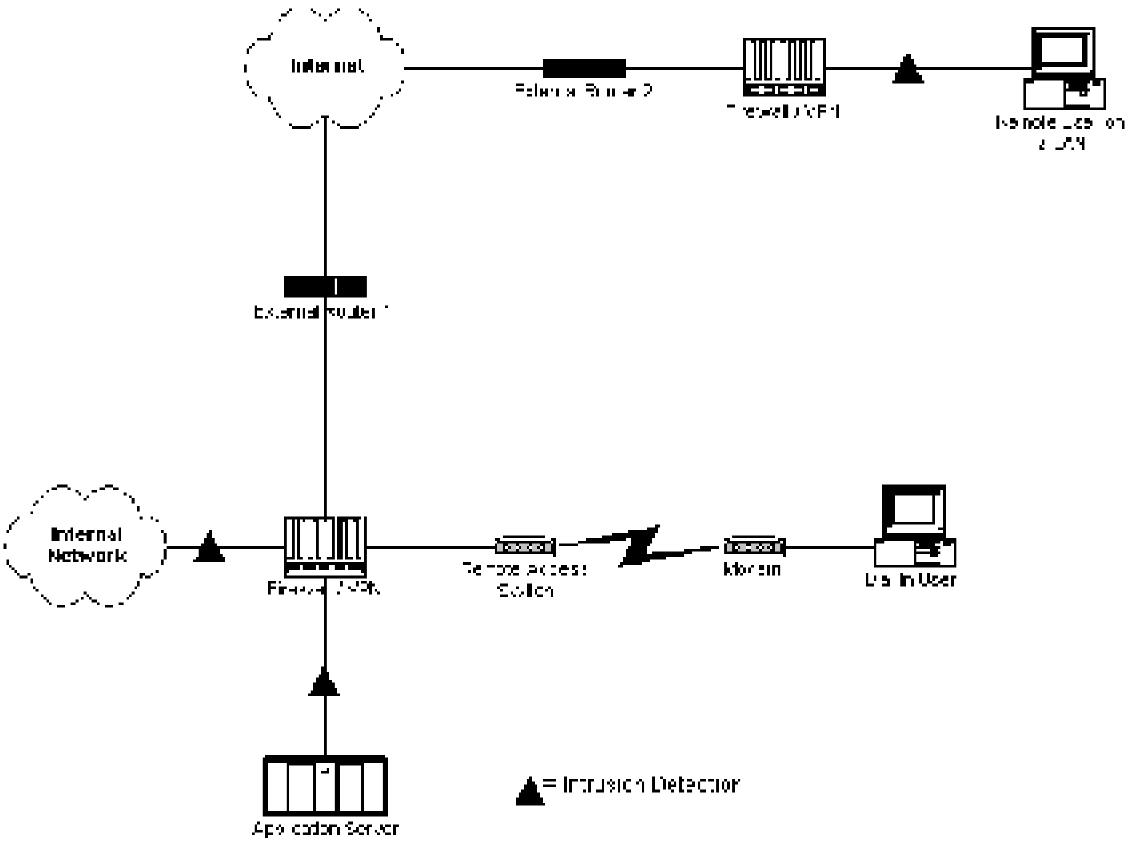
Scenario

You have a site that is providing an application to a distributed user base. Some are directly connected via a WAN and others dial-in. The application is a text-based application currently accessible via telnet. Suggest a redesign given the following diagram. Removing Dial-in is not an option.



Answer

The existing solution is obviously not very secure and if you hope to keep your application server and internal network from being attacked, you had best make some changes. I'll assume that for whatever reason that moving the application from telnet is not possible. The following diagram is one possible means of securing this architecture.



Perimeter Routers

Both Router 1 and Router 2 will have ingress and egress filters placed on them. The access list should include the limitation of only allowing VPN traffic to pass between Router 1 and Router 2. For simplicity I will use this key:

S0 router 1 will be 1.1.1.1
S0 router 2 will be 2.2.2.2
Eth0 will be 3.3.3.3
Firewall/VPN 1 will be 4.4.4.4
Firewall/VPN 2 will be 5.5.5.5

Sample access lists are as follows:
Ingress Filters for Router 1:

```
access-list 101 permit udp host 2.2.2.2 host 4.4.4.4 isacamp
access-list 101 permit esp host 2.2.2.2 host 4.4.4.4
access-list 101 deny any any
interface s0
ip access-group 101 out
```

Ingress Filters for Router 2:

```
access-list 102 permit udp host 1.1.1.1 host 5.5.5.5.isacamp
access-list 102 permit esp host 1.1.1.1 host 5.5.5.5
access-list 102 deny any any
interface s0
ip access-group 101 out
```

Egress Filters:

```
access-list 104 permit ip 3.3.3.0 0.0.0.255 any
interface eth0
ip access-group 104 in
```

Firewalls

Firewall 1 will have a rule base that only allows the internal network to communicate outbound to the Internet. It will also allow the internal network to communicate to the application server via telnet only. Another rule at the firewall will prevent the application server from initiating any communication. Firewall 2 will allow only those permitted clients to go out via the VPN.

Dial-In

The original design gave us a back door into the network via the application server. Since we do not want these users to come in un-logged right into our application server, I have moved the remote access server to it's own segment off of the firewall. By doing this, I can limit the services that can come in via dial-up, add another layer of login challenge, and specify where they can go via the firewall. The use of dial-back could also be used to help prevent attack. Another consideration would be to use a VPN client with these users providing they have access to the Internet.

Intrusion Detection

Intrusion Detection has been added to further layer our defensive architecture. There are still exploits that can be run on the application server such as grabbing the /etc/passwd file. To this end, IDS has been put in front of the application servers to look for patterns of attacks. The placement of this IDS will help us to react to alarms that need immediate action.

The IDS system that is placed in the remote site will help to prevent attack from being launched from that side and then tunneled through the VPN.

The final IDS will be placed on the internal network segment to help prevent our internal users from launching attacks towards the outside and the DMZ. As budgets allow, more Intrusion Detection systems should be placed near or on critical systems. The focus of this drawing was to protect our internal network and our DMZ from external attack.