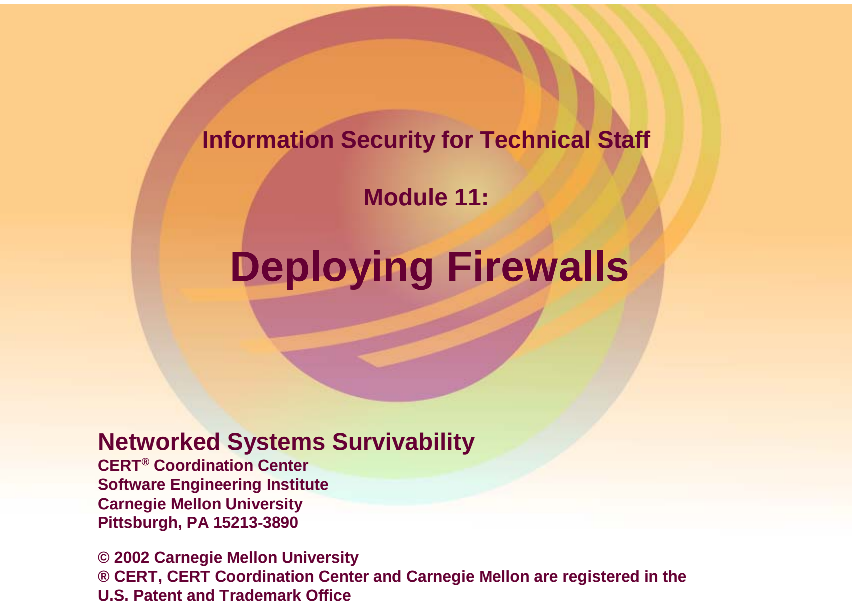


Information Security for Technical Staff

Module 11:

Deploying Firewalls

Networked Systems Survivability
CERT® Coordination Center
Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213-3890

Increasingly, organizations are connecting to the Internet to establish a business and electronic commerce presence and to access information rapidly. When your organization's networks are connected to the Internet without adequate security measures in place, you become vulnerable to attacks from external adversaries. Without firewalls, you will be unable to prevent many forms of undesirable access to your networks, systems, and information assets. The risks include:

- Loss of confidentiality of business information (e.g., financial records, strategic planning data, engineering models and prototypes, marketing plans, medical records, as well as inability to guarantee the integrity of such information)
- Loss of availability of mission-critical services such as EDI (electronic data interchange), ERP (enterprise resource planning), just-in-time inventory controls, and electronic mail
- Exposure of critical data about your information infrastructure that can be used by your adversaries in planning their attacks
- Legal liability, regulatory liability, or public loss of confidence when your adversaries use one of your computers to carry out attacks against other organizations
- Vandalism of public information services (such as your public Web site)

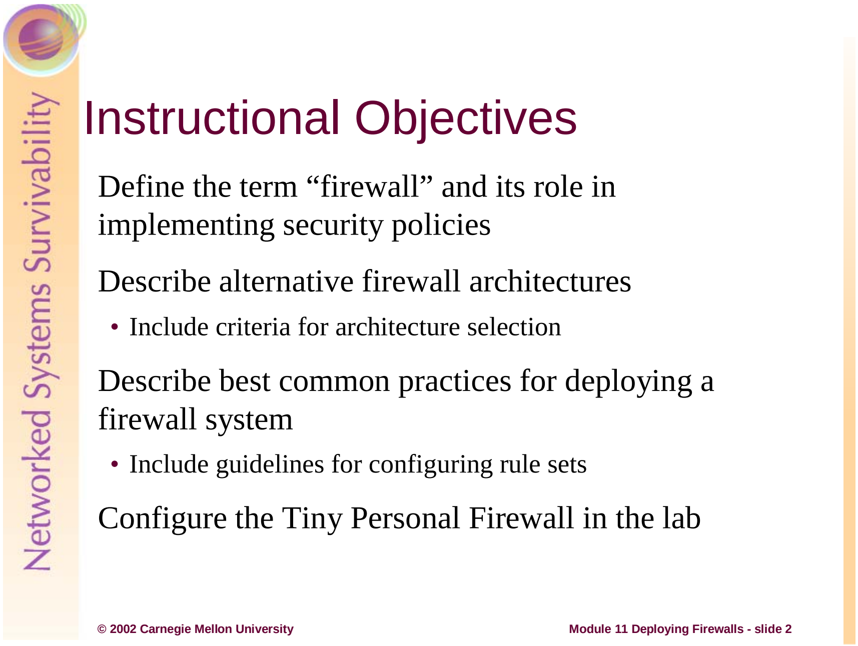
The use of firewall technology provides you with one of the most effective tools available to manage your networks' risk by providing you with access control mechanisms that can implement complex security policies [Allen 01].

According to ICSA (International Computer Security Association), 70% of sites with certified commercial firewalls are still vulnerable to attacks due to misconfiguration or improper deployment. January, 1999 [NetIQ 01].

Here's a list of 9 common administrative and social pitfalls related to firewalls. Doing any of these just about guarantees firewall-related security problems [ICSA 01].

1. Deploy a firewall without a security policy. This is fairly common, and includes those who have a policy that is old and not relevant anymore. Come up with a basic one, use it, stick to it, and work on a better one.

2. Circumvent the firewall security and appropriate use policy. This is the equivalent of propping the back door open. If the security policies need changing, then go through the proper steps to change them. The firewall should match an organization's security policy. The policy should not be ignored.
3. Add other services because users say they need them. A clear distinction between business "requirements" and "wants or desires" is very important. All business requirements related to the firewall (supported services, level of user accountability and authentication, etc.) should be weighed against a corresponding risk assessment for providing (and not providing) the service.
4. Concentrate on the firewall while ignoring other security measures. Firewalls are not enough. Some organizations still have a security checklist that has the word "firewall" next to the word "security" – with a large checkmark next to it. Firewalls are only one element of a defense-in-depth or layered security architecture.
5. Ignore the log files. Your security policy should address this. What are you going to do with the log files from your firewall? Some person or process should at least skim them, looking for anomalies. There are commercial products one can use to help with this.
6. Turn off those annoying warnings. Firewalls generate warning messages for a reason. It is illegal to disable a fire alarm in your office building, even if it sometimes interrupts your work. It should be considered equally bad to disable alarms on your firewall. It negatively impacts your network security perimeter.
7. Allow users on the firewall system. Most computer break-ins are through exploited user accounts. It should go without saying, but unfortunately does not, that you should never have user accounts on a firewall. Every user account is a potential avenue of attack. Every user is a potential attacker. Every keystroke of every user has the potential for opening a breach in the firewall through user error.
8. Allow a lot of people to administer the firewall. Too many cooks can spoil the broth. The same goes for firewalls. Every system administrator is a potential attacker – and usually can do more damage than a user.
9. Allow dial-in modems on desktops. Each is a hole in your security perimeter managed by the individual user. Every dial-in modem behind the firewall circumvents the security perimeter. Every dial-in modem inside the firewall perimeter is potentially an unguarded entrance to the organization's network.



Networked Systems Survivability

Instructional Objectives

Define the term “firewall” and its role in implementing security policies

Describe alternative firewall architectures

- Include criteria for architecture selection

Describe best common practices for deploying a firewall system

- Include guidelines for configuring rule sets

Configure the Tiny Personal Firewall in the lab

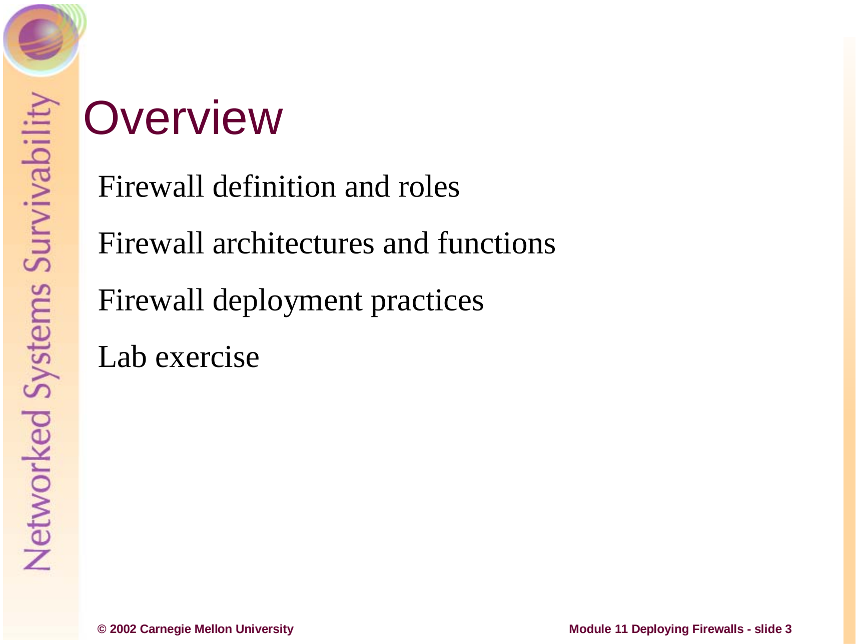
Module 11 Deploying Firewalls - slide 2

This module defines what is meant by the term “firewall” and “firewall system” and the various roles that a firewall can play in protecting systems and networks from unauthorized access – from both external and internal users.

There is a range of alternatives for architecting a firewall solution. An organization’s firewall architecture should derive from business objectives, requirements, and policy.

This module provides brief descriptions of seven recommended practices for deploying a firewall, including the case where a new firewall is replacing a currently operational firewall. Details of these practices can be found in [Allen 01].

This module concludes with a lab exercise using the Tiny Personal Firewall software. The exercise will demonstrate the concepts discussed in the slide and student workbook materials.



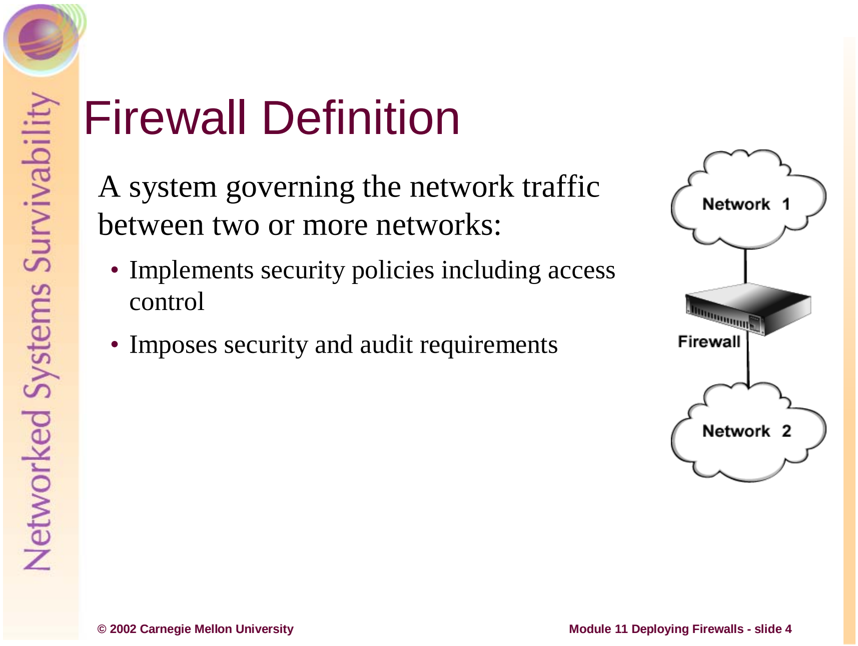
Networked Systems Survivability

Overview

- Firewall definition and roles
- Firewall architectures and functions
- Firewall deployment practices
- Lab exercise

Module 11 Deploying Firewalls - slide 3

This slide identifies the four major sections of this module, serving as a top-level table of contents.



Networked Systems Survivability

Firewall Definition

A system governing the network traffic between two or more networks:

- Implements security policies including access control
- Imposes security and audit requirements

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 4

The purpose of a firewall or firewall system (which comprises one or more hosts performing specific functions) is to serve as one element of an organization’s perimeter defense. The perimeter can be defined as what separates the external world from the internal network or what separates internal sub-networks with differing access requirements. Ultimately, a firewall implements policy that specifies how network traffic is to move between two or more networks.

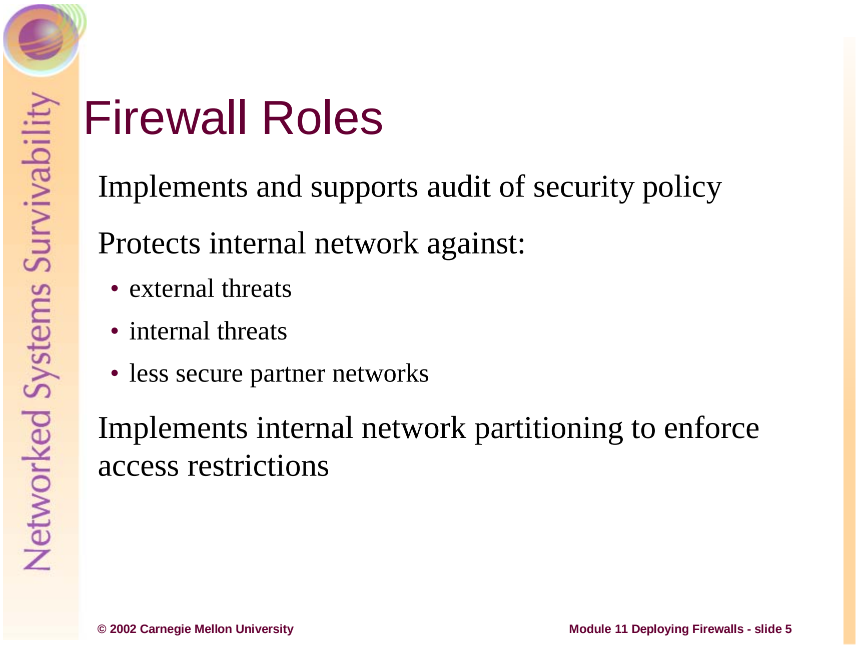
A firewall intercepts and controls traffic between networks with differing levels of trust – different security domains. A firewall is an excellent place to focus security decisions and to enforce a network security policy. In addition, firewalls can often serve as a single location where inter-network activity can be efficiently recorded/logged [ICSA 01].

If a business purchases a firewall when it sets up an Internet connection, but has no policies, standards, or architecture, it will be difficult for the administrators to make decisions on how to configure the firewall [Ranum 98].

A firewall is not intended to address the following questions. Other protection mechanisms are required.

- Does the organization permit or prohibit modems on desktops that connect directly to the Internet, bypassing the firewall? If permitted, how is protection ensured?
- How does the organization address viruses that pass through the firewall as part “permitted” files, such as email attachments?
- How does the organization deal with mobile code – Java and ActiveX?
- In general, a firewall cannot protect against a data-driven attack – attacks in which something is mailed or copied to an internal host where it is then executed [Ranum 98, Curtin 00].

For further information, refer to [Ranum 98, Curtin 00], [Allen 01], and [Lynch 00].



Networked Systems Survivability

Firewall Roles

Implements and supports audit of security policy

Protects internal network against:

- external threats
- internal threats
- less secure partner networks

Implements internal network partitioning to enforce access restrictions

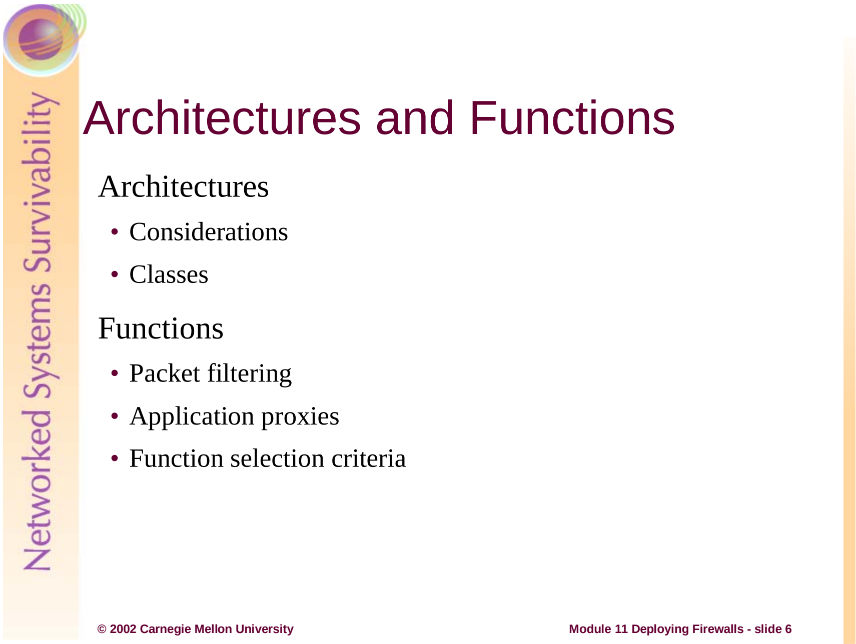
© 2002 Carnegie Mellon University Module 11 Deploying Firewalls - slide 5

A firewall is a highly desirable "choke point" through which all traffic flows. As a result, it serves as a logical point for monitoring for policy compliance, examining network traffic flow and performance, detecting signs of suspicious or unexpected behavior, capturing detailed log information for later analysis, and implementing alerts for high-priority action.

A firewall is most frequently deployed to protect an organization's internal networks from the outside world, such as the Internet. It does so by blocking or denying both incoming and outgoing traffic that is not permitted by policy. Firewall rules and configuration need to be reviewed on a regular basis as attack patterns change frequently and new vulnerabilities are discovered almost daily.

In today's business climate, organizations frequently initiate and terminate business partnerships and alliances. Providing access to enable timely flow of information and transactions is often critical to partnership success. But not all organizations manage their networks with the same level of security. So it is prudent to use a firewall to restrict access and protect your networks when connecting to those belonging to your partners. The same holds true when acquiring new companies and organizational units.

In addition, a firewall can be effectively used when access to hosts and data on internal sub-networks needs to be controlled to protect sensitive information and restrict its distribution. Where two sub-networks connect, each having differing access requirements, a firewall can be used to implement these requirements.



Networked Systems Survivability

Architectures and Functions

Architectures

- Considerations
- Classes

Functions

- Packet filtering
- Application proxies
- Function selection criteria

© 2002 Carnegie Mellon University

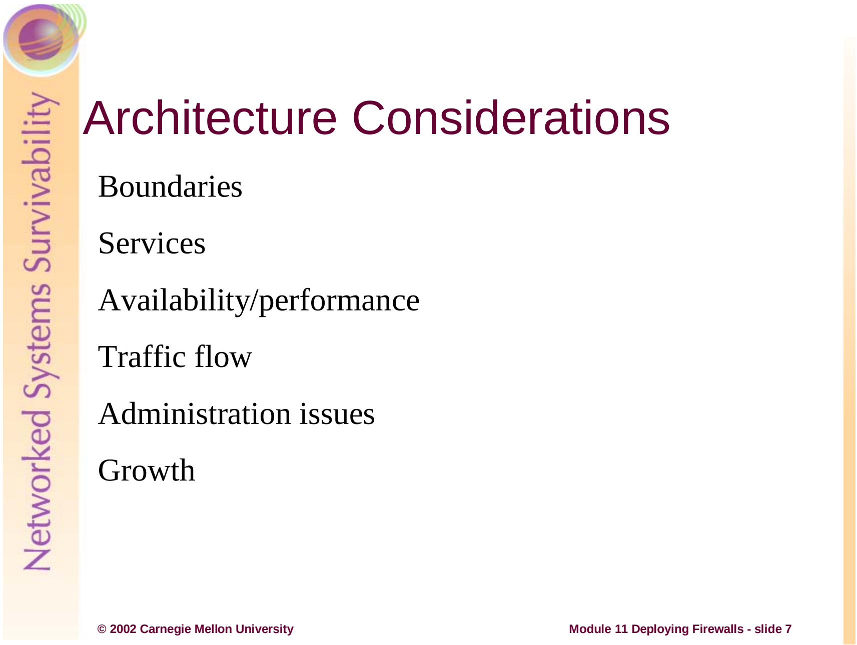
Module 11 Deploying Firewalls - slide 6

This slide introduces and serves as a table of contents for the next section on architecture classes and functions.

Architecture considerations include those requirements and policies you need to take into account when selecting a firewall architecture. Architecture classes include single and multi layer. A single layer architectural approach allocates all firewall functions to a single host. A multi layer architecture distributes firewall functions amongst 2 or more hosts.

Packet filtering is a firewall function that examines and makes permit/deny decisions based on packet header information and, in some cases, on the state of the protocol session. Application proxies provide a greater level of security than packet filtering as they can operate on both packet header and content.

This section closes with a brief discussion of how to go about selecting firewall functions that meet your security policies and requirements.



Networked Systems Survivability

Architecture Considerations

Boundaries

Services

Availability/performance

Traffic flow

Administration issues

Growth

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 7

Designing a firewall requires that you understand and identify the boundaries between security domains in your network. A network security domain is a contiguous region of a network that operates under a single, uniform security policy and typically under the same administrative control. Wherever these domains intersect, there is a potential need for a policy conflict resolution mechanism at that boundary such as a firewall.

Your organization's security policy should address:

- the objective that all incoming and outgoing network traffic must go through the firewall (that is, no traffic which bypasses the firewall is permitted, for example, by using modems) — or conversely, that specific loopholes are permitted and under what conditions (such as through the use of modems, tunnels, virtual private networks (VPN), connections to Internet Service Providers (ISP))
- the services you intend to offer to untrusted networks from your protected network. These could be offerings to the Internet or to other internal networks.
- the services you intend to request from untrusted networks via your protected network. These could be requests to the Internet or to other internal networks.

Firewall systems provide a policy enforcement mechanism at a security domain boundary. If an adversary can exploit another less protected boundary to gain access into your network (for example, a modem on a user workstation or via a partner’s network), then any firewall systems you have deployed on other boundaries to control access to that network will be ineffective.

In the offering and requesting of services, your policy should ensure that you only allow network traffic that is determined to be safe and in your interests and that minimizes the exposure of information about your protected network’s information infrastructure [Allen 01].

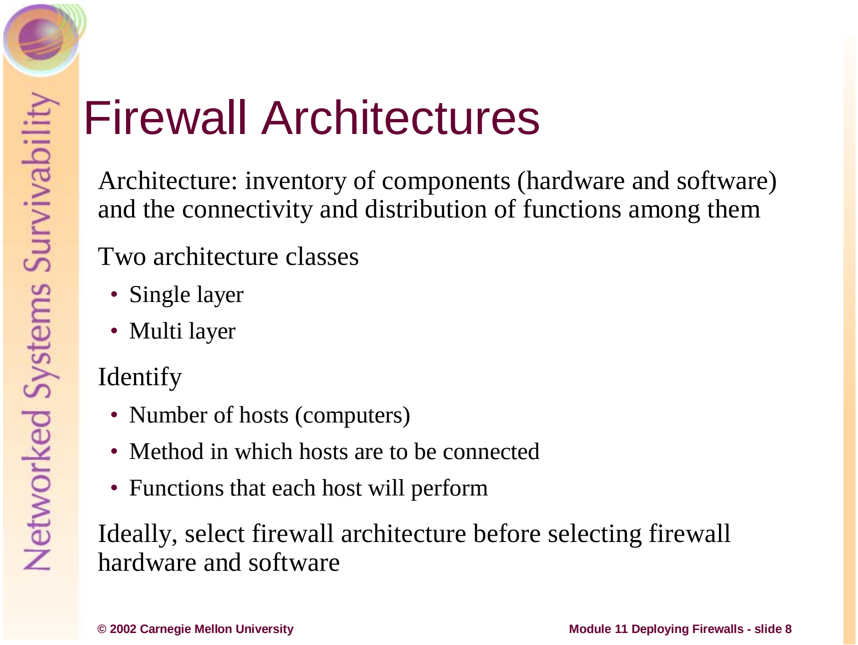
There is an inverse relationship between the number of interfaces a firewall supports and firewall performance. As interfaces grow, having multiple firewalls may be preferable to adding more interfaces to a single firewall. You need to understand as best you can what traffic loads the firewall will need to handle and your strategy for dealing with firewall downtime.

There is a significant level of trust in granting access to the firewall system for administration purposes so this needs to be kept to the smallest number of administrators possible. In addition, you need to determine if the firewall can only be administered locally using the firewall console or if remote (and secure) administration will be permitted, particularly during non-business hours. Remote administration requires the use of strong authentication and encryption technologies.

Replacing an operational firewall can be complex, time consuming, disruptive, and expensive, so attempt to identify your future needs for additional firewall capacity to accommodate growth and change. Always acquire more capacity than you can anticipate as needs will expand to use up that capacity.

Additional architectural considerations include:

- Defining your required levels of firewall monitoring, redundancy, and control
- Identifying costs for purchase, system management, and maintenance, both one time and recurring



Networked Systems Survivability

Firewall Architectures

Architecture: inventory of components (hardware and software) and the connectivity and distribution of functions among them

Two architecture classes

- Single layer
- Multi layer

Identify

- Number of hosts (computers)
- Method in which hosts are to be connected
- Functions that each host will perform

Ideally, select firewall architecture before selecting firewall hardware and software

© 2002 Carnegie Mellon University Module 11 Deploying Firewalls - slide 8

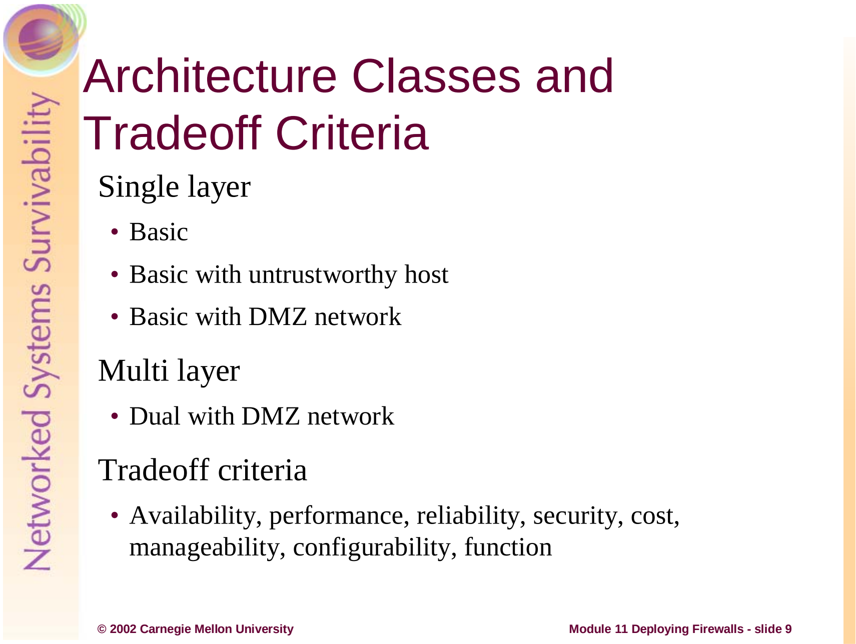
The most common boundary where firewalls are applied today is between an organization’s internal networks and the Internet. When establishing an Internet firewall, the first thing you must decide is its basic architecture. This assumes you have previously established your firewall requirements and the security policy it is intended to implement. In this context, architecture refers to the inventory of components (hardware and software), and the connectivity and distribution of functions among them.

There are two classes of firewall architectures, which we refer to as single layer and multiple layer. In a single layer architecture, one network host is allocated all firewall functions and is connected to each network for which it is to control access. This approach is usually chosen when containing cost is a primary factor or when there are only two networks to interconnect. The advantage of this approach is that everything there is to know about the firewall resides on that one host. In cases where the policy to be implemented is simple and there are few networks being interconnected, this approach can also be very cost-effective to operate and maintain over time. The greatest disadvantage of the single layer approach is its susceptibility to implementation flaws or configuration errors — depending on the type, a single flaw or error might allow firewall penetration.

In a multiple layer architecture, the firewall functions are distributed among a small number of hosts, typically connected in series, with demilitarized zone (DMZ) networks between them. This approach is more difficult to design and operate, but can provide substantially greater security by diversifying the defenses you are implementing. Although more costly, we advise using different technology in each of these firewall hosts. This reduces the risk that the same implementation flaws or configuration errors will exist in every layer. The most common design approach for this type of architecture is an Internet firewall composed of two hosts interconnected with one DMZ network [Allen 01].

Firewall systems can be implemented using a number of hardware and software solutions including:

- application-type firewalls that run on widely used operating systems (flavors of Unix, Linux, Windows) on a standard host or server
- appliance-type firewalls that run on proprietary operating systems, typically on proprietary hardware, that are developed solely for this purpose. Examples include Cisco's SecurePix and NetScreen. These can potentially be more secure out of the box, as the operating system and hardware are hardened as part of the design


Networked Systems Survivability

Architecture Classes and Tradeoff Criteria

Single layer

- Basic
- Basic with untrustworthy host
- Basic with DMZ network

Multi layer

- Dual with DMZ network

Tradeoff criteria

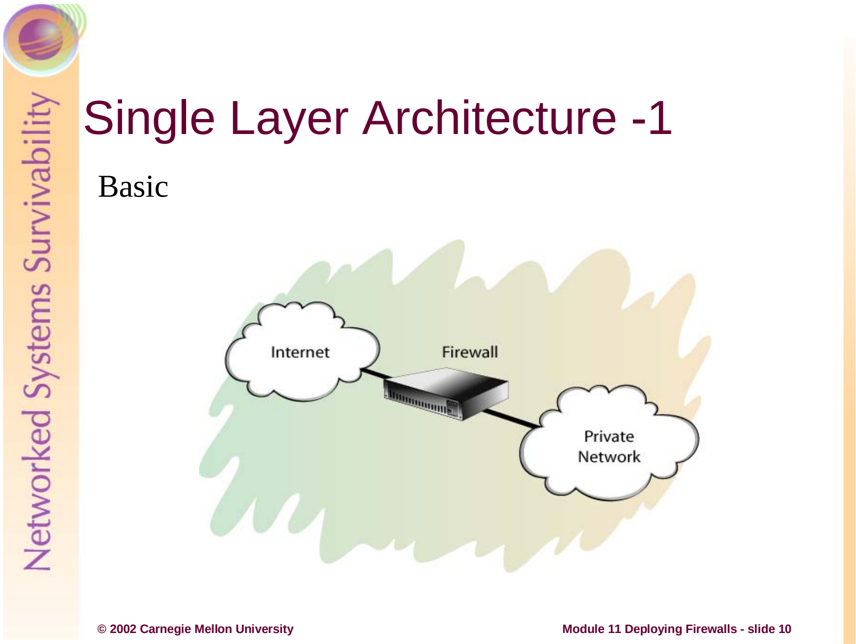
- Availability, performance, reliability, security, cost, manageability, configurability, function

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 9

Single layer and multi layer architecture classes are described in more detail on the next four pages. You must make the same architectural tradeoffs in designing your firewall that are commonly made in other mission-critical systems. Some significant tradeoff criteria include:

- **availability**. Availability is achieved by a combination of reliability and redundancy. Start by choosing hardware and software components that are reliable, and add redundant components as needed.
- **performance**. Based on the anticipated traffic through the firewall system, you may need multiple firewall hosts to distribute the load and handle traffic at an acceptable rate.
- **security**. Weigh the use of single versus dual firewall systems at your network perimeter. The factors to consider include
  - having outside traffic passing through two firewall systems instead of one (benefits vs. cost)
  - your ability to monitor traffic and the monitoring locations
  - your ability to recover from compromises including disconnecting one firewall system while keeping the other operational
  - your needs for and number of network ports
  - performance
  - failure characteristics
  - expense
  - complexity of firewall system operations and maintenance
  - using multiple firewall systems from different vendors to reduce your exposure to vulnerabilities inherent in a single product (survivability through diversity)



Networked Systems Survivability

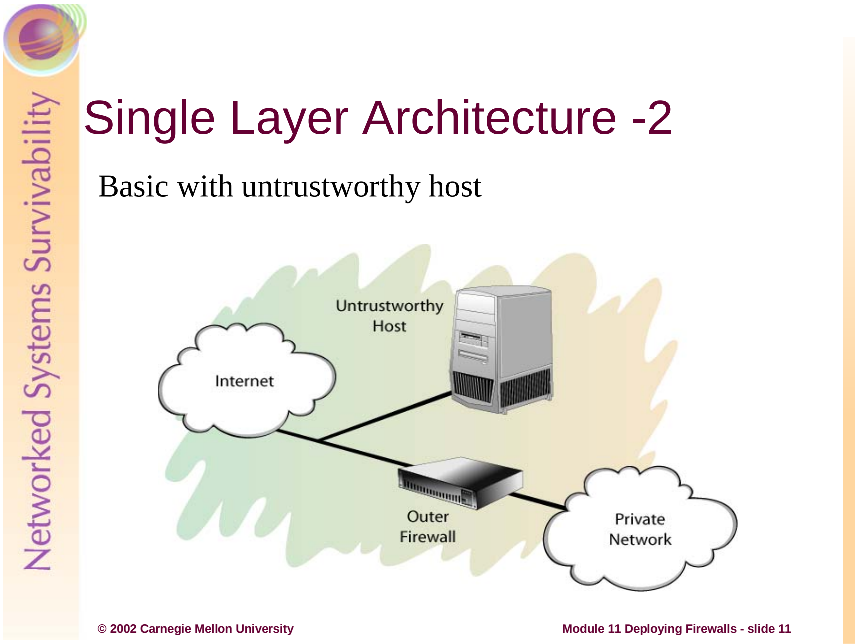
Single Layer Architecture -1

Basic

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 10

This is the starting point for all firewalls. A basic border firewall is a single host interconnecting an organization’s internal network and some untrusted network, typically the Internet. In this configuration, the single host provides all firewall functions.



Networked Systems Survivability

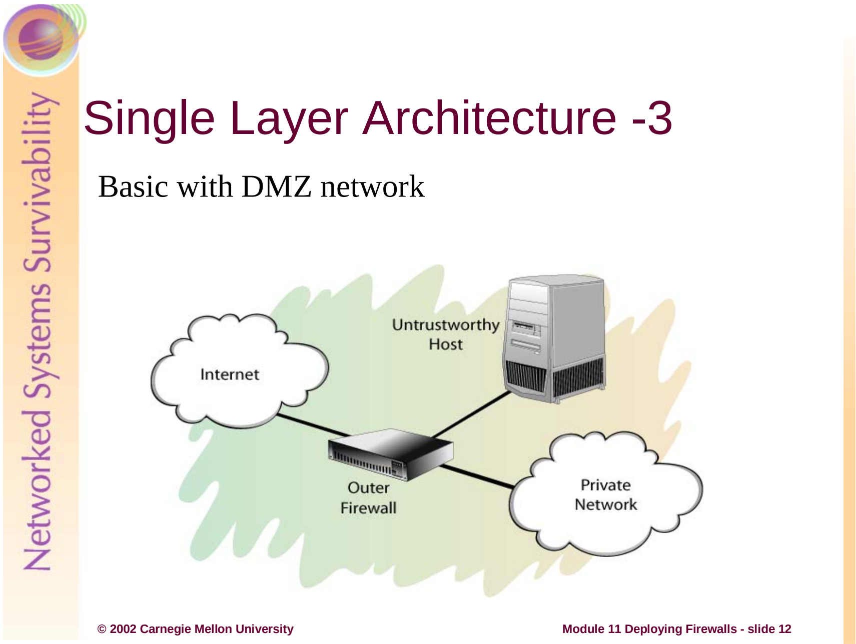
Single Layer Architecture -2

Basic with untrustworthy host

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 11

To the basic border firewall, add a host that resides on an untrusted network where the firewall cannot protect it. That host is minimally configured and carefully managed to be as secure as possible (sometimes referred to as a bastion host). The firewall is configured to require incoming and outgoing traffic to go through the untrustworthy host. The host is referred to as untrustworthy because it cannot be protected by the firewall; therefore, hosts on the trusted networks can place only limited trust in it.



Networked Systems Survivability

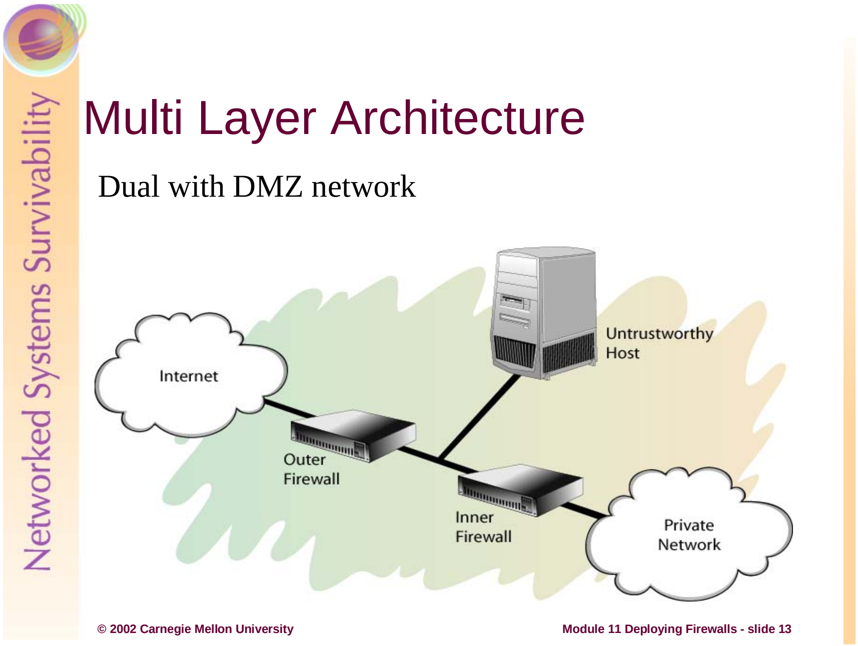
Single Layer Architecture -3

Basic with DMZ network

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 12

In a DMZ network, the untrusted host is brought “inside” the firewall, but placed on a network by itself (the firewall host then interconnects three networks). This increases the security, reliability, and availability of the untrusted host, but it does not increase the level of trust that other “inside” hosts can afford it. Other untrustworthy hosts for other purposes (for example, a public web site or ftp server) can easily be placed on the DMZ network, creating a public services network.



Networked Systems Survivability

Multi Layer Architecture

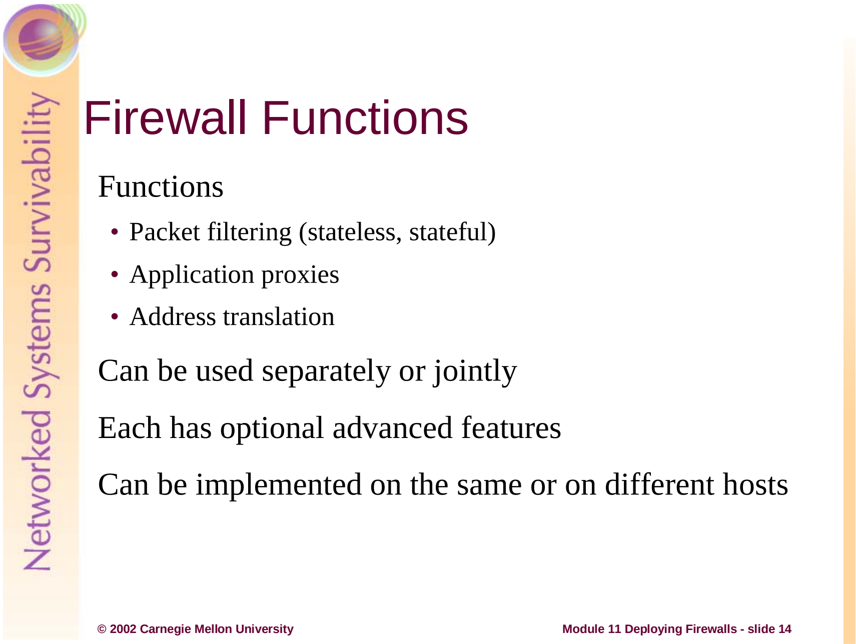
Dual with DMZ network

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 13

The organization’s internal network is further isolated from the untrustworthy network by adding a second firewall host. By connecting the untrustworthy network to one firewall host, the organization’s internal network to the other, and the DMZ between, traffic between the internal network and the Internet must traverse two firewalls and the DMZ.

In each of these architectures, firewalls are used to control access at the border of your network mainly for the purpose of protecting your network from an untrusted network. Firewalls deployed entirely within your network can also be used to provide mutual protection among subnets of your network. Controlling access between internal subnets is no different than controlling access between your network and the Internet, so all of the above architectures can be used as internal firewall architectures as well.



Networked Systems Survivability

Firewall Functions

Functions

- Packet filtering (stateless, stateful)
- Application proxies
- Address translation

Can be used separately or jointly

Each has optional advanced features

Can be implemented on the same or on different hosts

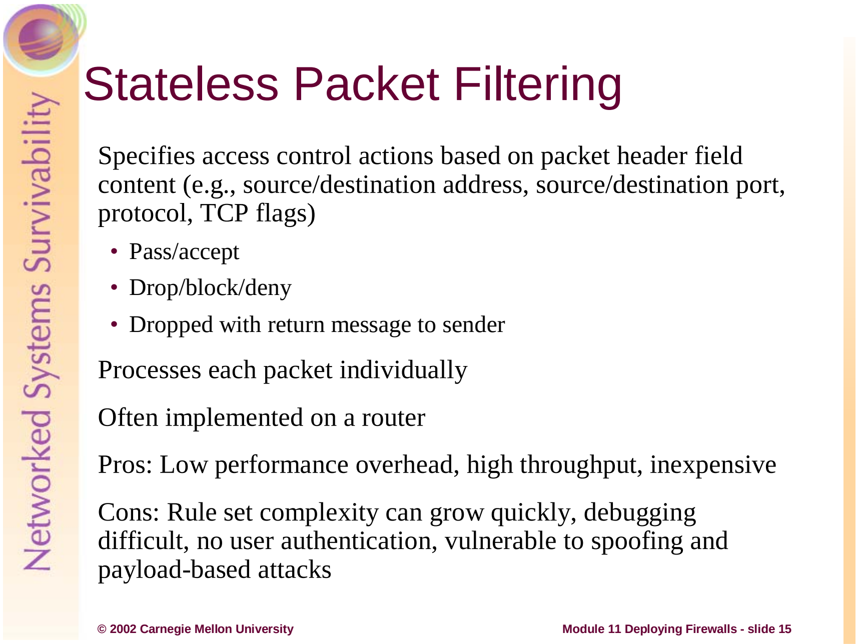
Module 11 Deploying Firewalls - slide 14

Having chosen the basic architecture (that is, the number of hosts, the method in which they are connected, the tasks that each will perform), the next step is to select the firewall function(s) to be implemented in these hosts. The two most basic categories of firewall function are packet filtering and application proxies. These functions can be used separately or jointly and can be on the same or on different firewall hosts.

There are good reasons to use both packet filtering and application proxies. Certain services (such as SMTP, HTTP, or NTP) are usually safe to control via stateless packet filters while others (such as DNS or FTP) may require the more complex features available only in stateful packet filters or proxies. Packet filtering is fast, while application proxies are generally slower. In cases where greater access control is required and the poorer performance of proxies cannot be tolerated, stateful (vs. stateless) packet filters may be an acceptable compromise. In any case, one should plan to have as many of these different functions available as possible, applying each where appropriate [Allen 01].

Stateless and stateful packet filtering, application proxies, and address translation are described in detail on the following pages.

For further information on firewall architectures and functions, refer to [Cheswick 94], [Zwicky 00], [Brenton 01], [Brenton 99], [Ogletree 00], and [Smith 01].


Networked Systems Survivability

Stateless Packet Filtering

Specifies access control actions based on packet header field content (e.g., source/destination address, source/destination port, protocol, TCP flags)

- Pass/accept
- Drop/block/deny
- Dropped with return message to sender

Processes each packet individually

Often implemented on a router

Pros: Low performance overhead, high throughput, inexpensive

Cons: Rule set complexity can grow quickly, debugging difficult, no user authentication, vulnerable to spoofing and payload-based attacks

© 2002 Carnegie Mellon University Module 11 Deploying Firewalls - slide 15

Some of the more common items packet filters can act upon are [Smith 01]:

- Source address (e.g., pass in all packets from 192.168.1.0 through 192.168.1.255 but all other packets are blocked)
- Destination address (e.g., packets bound for 128.162.11.14 are not permitted to pass)
- Source and destination port number (e.g., all TCP packets bound for port 80 [the HTTP port] would be permitted in but TCP packets bound for ports 137-139 [NetBIOS/NetBUI] would be blocked)
- Protocol type (e.g., TCP, UDP, ICMP, DECnet, IPX)
- The network interface through which the packet enters
- The direction of traffic (inbound or outbound)
- Source routing
- Fragmentation
- Connection state (e.g., SYN, SYN/ACK, FIN)

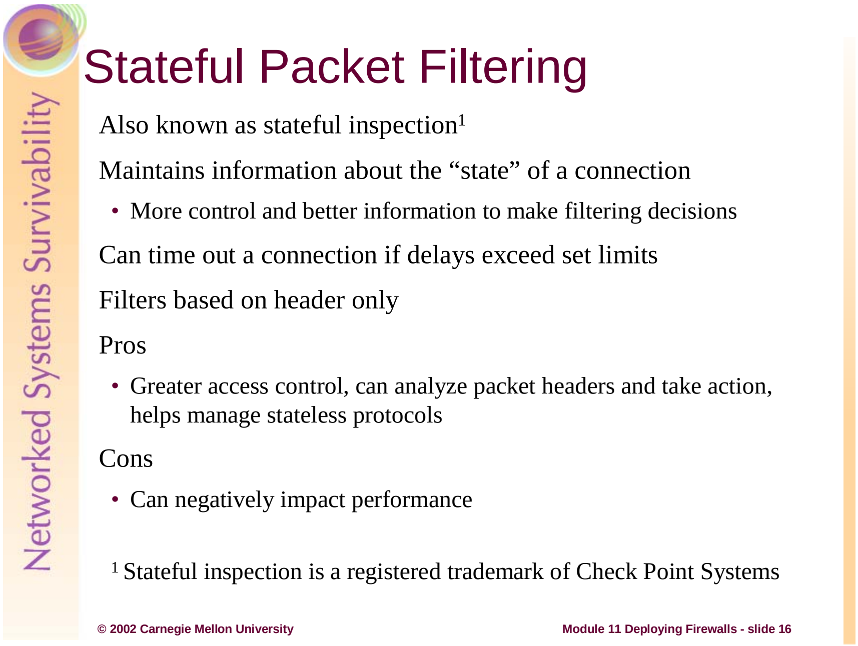
For example, the security policy says, "accept inbound email (this is SMTP on TCP port 25) from everybody but bigspammers.org." The firewall administrator would write up two rules. The first would drop any connection from bigspammers.org to the firewall. The second rule would pass from anybody on TCP port 25 to the firewall. The rules are applied in top-down order. Rules are usually ordered from most restrictive to least restrictive. If they were reversed, the rule dropping packets from big-spammers.org would not be reached because the less restrictive rule matched first and the packet was passed.

Early packet filtering firewalls provided low overhead, high throughput, was inexpensive, and provided good traffic management.

As the Internet evolved, it became apparent packet filtering firewalls had several disadvantages:

- Direct connections by external clients to internal hosts
- Packet filtering rules become unmanageable in complex environments
- Vulnerable to attacks such as IP Spoofing, i.e., impersonating another system by using its IP address
- No user authentication

The only effective means of controlling connectionless protocols such as UDP with a stateless packet filter is either to block the port or to let it through and hope for the best. Stateless packet filters can be applied to the Type and Code fields of ICMP messages [Brenton 99].



Networked Systems Survivability

Stateful Packet Filtering

Also known as stateful inspection[1]

Maintains information about the “state” of a connection

- More control and better information to make filtering decisions

Can time out a connection if delays exceed set limits

Filters based on header only

Pros

- Greater access control, can analyze packet headers and take action, helps manage stateless protocols

Cons

- Can negatively impact performance

[1] Stateful inspection is a registered trademark of Check Point Systems

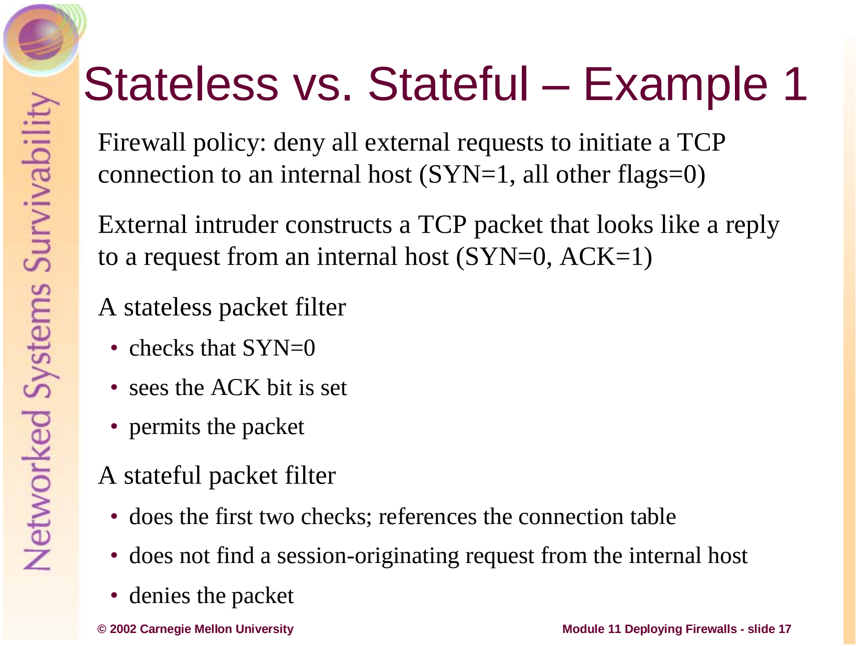
© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 16

Stateful packet filtering takes stateless packet filtering one step further by maintaining a connection table in order to monitor the state or context of a communication session by attempting to match up outgoing and incoming packets. The information retained in the table usually includes the source and destination addresses and source and destination ports. Stateful packet filtering does not simply rely on flag settings. Every time an external packet appears to be responding to an internal request, the connection table is referenced to ensure:

- The internal host actually initiated the request
- The source port matches the originating request
- The destination port matches the originating request

A stateful packet filter may even verify that the sequence and acknowledgment numbers all match. If all this data is correct, the stateful packet filter allows the packet to pass. Once the FIN packets are sent by each system (terminating a TCP session), the connection table entry is removed. Additionally, if no reply is received for a period of time (anywhere from one minute to one hour, depending on the configuration), the firewall assumes that the remote server is no longer responding and again deletes the connection table entry. This keeps the table current [Brenton 99], [Ogletree 01].



Networked Systems Survivability

Stateless vs. Stateful – Example 1

Firewall policy: deny all external requests to initiate a TCP connection to an internal host (SYN=1, all other flags=0)

External intruder constructs a TCP packet that looks like a reply to a request from an internal host (SYN=0, ACK=1)

A stateless packet filter

- checks that SYN=0
- sees the ACK bit is set
- permits the packet

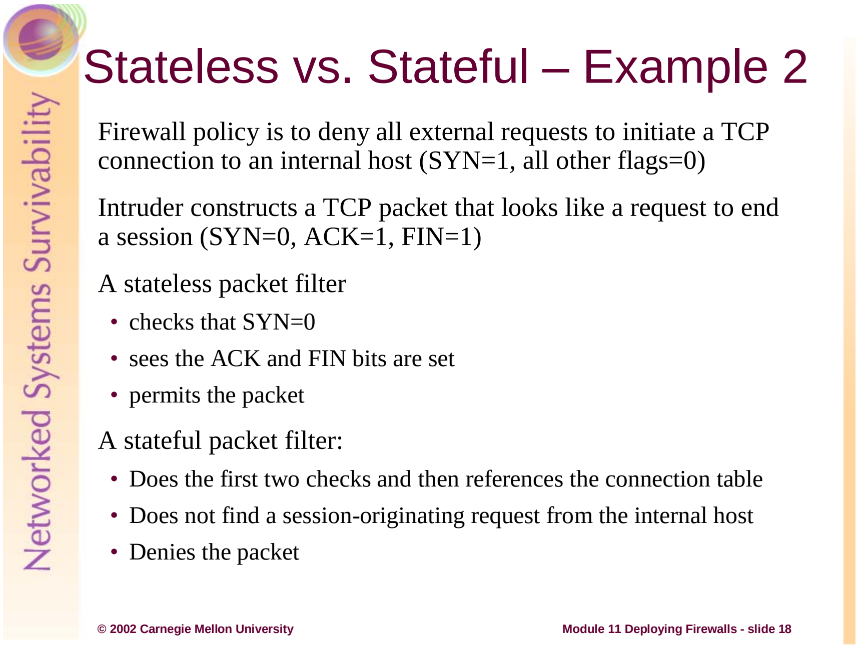
A stateful packet filter

- does the first two checks; references the connection table
- does not find a session-originating request from the internal host
- denies the packet

© 2002 Carnegie Mellon University Module 11 Deploying Firewalls - slide 17

Assume that an attacker sends an internal host a packet of data with a payload designed to crash the host. The attacker constructs the packet header to look like a reply to information requested by the internal host (SYN = 0, ACK = 1). A stateless packet filter analyzes this packet, sees that the ACK bit is set, and is fooled into thinking that this is a reply to a legitimate request initiated by the internal host. It permits the attacker's packet to be sent to the host.

A stateful packet filter references its connection table. While reviewing the table, the stateful packet filter realizes that the internal host never actually connected to this external system to place a data request. Since this information has not been explicitly requested, the stateful packet filter blocks the packet and it is dropped [Brenton 99].


Networked Systems Survivability

Stateless vs. Stateful – Example 2

Firewall policy is to deny all external requests to initiate a TCP connection to an internal host (SYN=1, all other flags=0)

Intruder constructs a TCP packet that looks like a request to end a session (SYN=0, ACK=1, FIN=1)

A stateless packet filter

- checks that SYN=0
- sees the ACK and FIN bits are set
- permits the packet

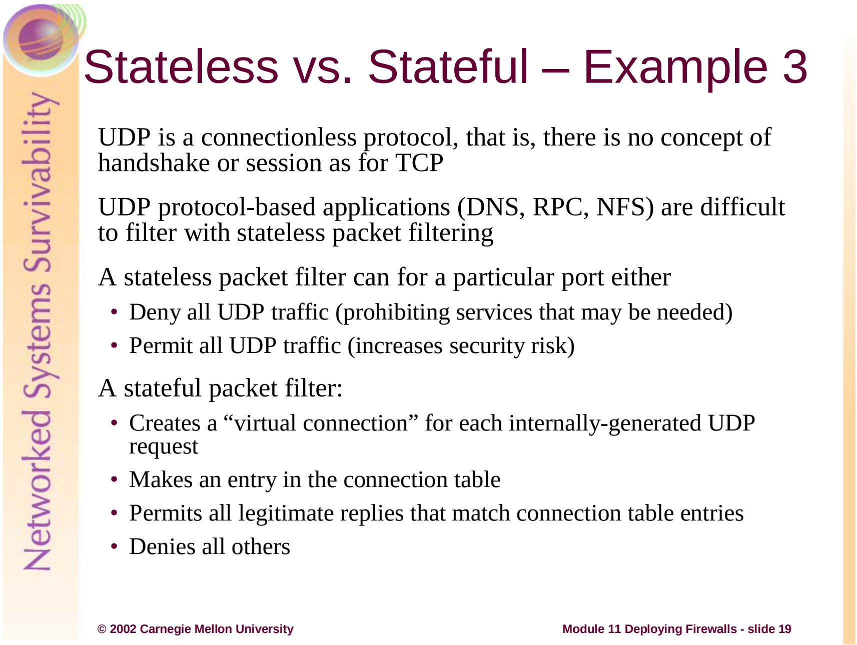
A stateful packet filter:

- Does the first two checks and then references the connection table
- Does not find a session-originating request from the internal host
- Denies the packet

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 18

The attacker attempts to perform a FIN scan by transmitting packets with the ACK and FIN bits set to 1 (SYN=0, ACK=1, FIN=1). A stateless packet filter permits this packet, assuming it is a request to end a session. A stateful packet filter recognizes that the SYN bit is not set and proceeds to compare this traffic to the connection table. It realizes that the internal host never initiated a session with the attacker’s system. As a result, there is no legitimate reason that the attacker should be trying to end the session. The stateful packet filter blocks the packet [Brenton 99].



Networked Systems Survivability

Stateless vs. Stateful – Example 3

UDP is a connectionless protocol, that is, there is no concept of handshake or session as for TCP

UDP protocol-based applications (DNS, RPC, NFS) are difficult to filter with stateless packet filtering

A stateless packet filter can for a particular port either

- Deny all UDP traffic (prohibiting services that may be needed)
- Permit all UDP traffic (increases security risk)

A stateful packet filter:

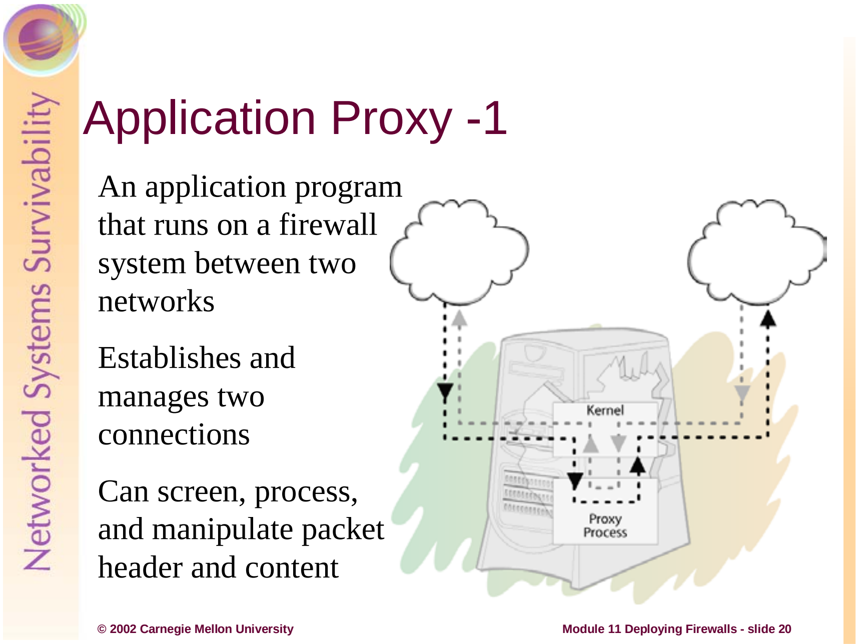
- Creates a “virtual connection” for each internally-generated UDP request
- Makes an entry in the connection table
- Permits all legitimate replies that match connection table entries
- Denies all others

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 19

Stateless protocols can be made more secure with stateful packet filtering. UDP protocol-based applications (DNS, RPC, NFS) are difficult to filter with stateless packet filtering because there is no concept of request and response, hence the term "stateless". With stateless packet filtering, beyond filtering on IP address and particular UDP port numbers, the choice is either to disallow all UDP-based traffic or open the communication channels and expose internal systems to security risks.

Stateful packet filtering secures UDP-based traffic by creating a virtual connection on top of the UDP communications. Each UDP request packet permitted to cross the firewall is recorded in the connection table. UDP packets traveling in the opposite direction are verified against the ones awaiting a response in connection table. A packet that is a legitimate response to a request packet is passed on and all others are dropped. If a response does not arrive in a specified period of time, the connection is timed out. Thus, even UDP applications can be secured [Smith 01].



Networked Systems Survivability

Application Proxy -1

An application program that runs on a firewall system between two networks

Establishes and manages two connections

Can screen, process, and manipulate packet header and content

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 20

As an analogy, think of two people speaking through a language interpreter. While it is true these two people are carrying on a conversation, they never actually speak to one another. All communication passes through the interpreter before being passed on to the other party. The interpreter might have to clean up some of the language used, or filter out comments or statements that might seem hostile [Brenton 99].

An application proxy is an application program that runs on a firewall system between two networks. It understands the service protocol that it is responsible for processing, it implements protocol/service-specific security such as access control and levels of authentication, and makes all packet-forwarding decisions. Application-level proxy servers evaluate the request and decide to permit or deny based on a set of rules that apply to the individual network service (e.g., SMTP, HTTP, FTP, etc.) as well as host/user permissions. Proxy servers mirror the service as if it were running on the end host [Smith 01].

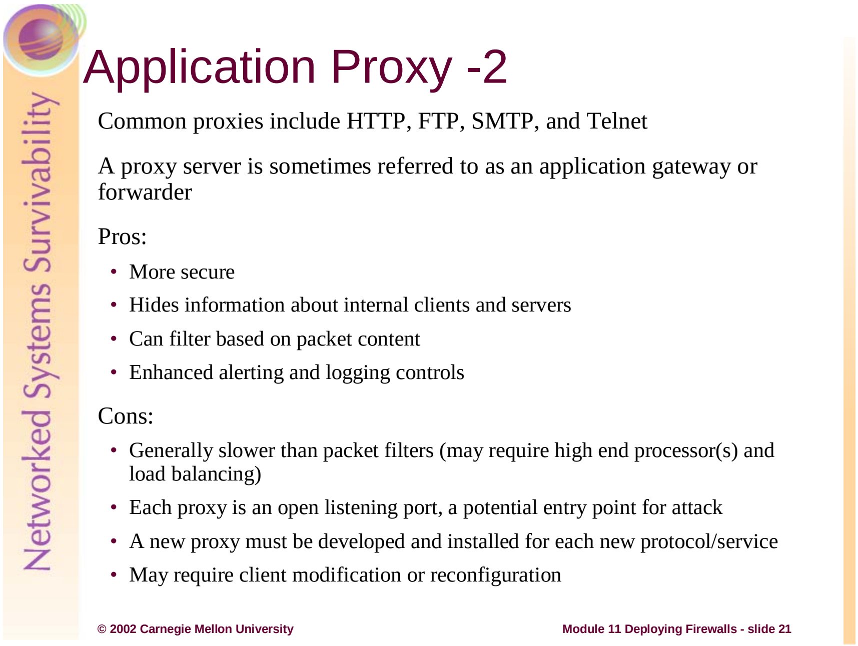
An application proxy establishes and manages two connections: one between the requestor and the proxy server and one between the proxy server and the destination service.

Application proxy-based firewalls operate at the application level and can examine information at the application data level. They are not limited to per-packet data, but can examine the entire data stream (since they are not dealing with packets). They can make their decisions based on application data, such as commands passed to FTP, or a URL passed to HTTP [ICSA 01].

Proxy servers also can [Smith 01] [Brenton 99]:

- Generate detailed audit records and session information
- Perform data modification
- Provide levels of authentication such as
    - a client system that could host multiple users or services
    - a specific session/service request, requiring that each new request be authenticated
    - a user by user basis
- Perform URL filtering, and caching
- Block questionable packet content

Using a proxy to block Java and ActiveX typically blocks both good and bad code, without distinguishing between the two (all or nothing).



Networked Systems Survivability

Application Proxy -2

Common proxies include HTTP, FTP, SMTP, and Telnet

A proxy server is sometimes referred to as an application gateway or forwarder

Pros:

- More secure
- Hides information about internal clients and servers
- Can filter based on packet content
- Enhanced alerting and logging controls

Cons:

- Generally slower than packet filters (may require high end processor(s) and load balancing)
- Each proxy is an open listening port, a potential entry point for attack
- A new proxy must be developed and installed for each new protocol/service
- May require client modification or reconfiguration

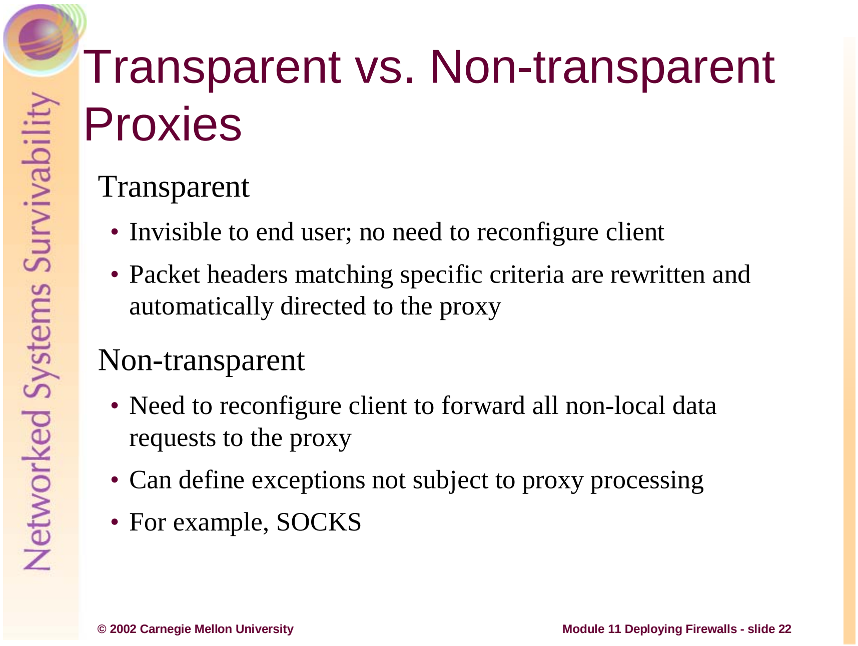
© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 21

Default proxy software often is part of a firewall configuration. Proxies for common protocols such as HTTP, FTP, SMTP, and Telnet are typically included.

Proxies provide a greater level of security by ensuring that two connecting hosts never exchange packets directly. Given they operate at the application level, they can filter based on packet content, and provide a central point for more sophisticated and relevant alerts and logging information.

Proxy services can require significant computing resources and a high-end host. Each proxy listens on an open port for its specific protocol. Open ports can be vulnerable to attack. Proxy servers are application specific. In order to support a new protocol via a proxy, a proxy must be developed for that protocol. If you select a proxy firewall, you need to ensure that it supports all of the applications requiring proxy service. Non-transparent proxies (see next page) may require that special software be installed on each client. You need to factor in configuration and maintenance requirements when using non-transparent proxies.


Networked Systems Survivability

Transparent vs. Non-transparent Proxies

Transparent

- Invisible to end user; no need to reconfigure client
- Packet headers matching specific criteria are rewritten and automatically directed to the proxy

Non-transparent

- Need to reconfigure client to forward all non-local data requests to the proxy
- Can define exceptions not subject to proxy processing
- For example, SOCKS

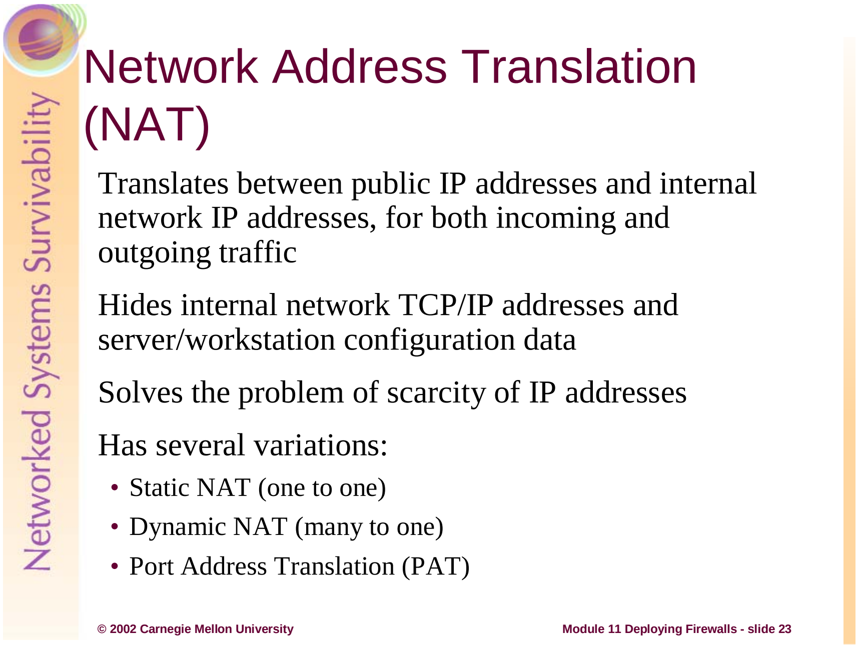
© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 22

Transparent proxies can be configured to be totally invisible to the end user. A transparent proxy combines packet filtering, packet rewriting, and traditional application proxies. All internal hosts are configured as though the proxy were a regular router leading to the Internet. Selected fields (e.g., destination) of packet headers that match specific criteria are rewritten so that they are directed to the proxy. Packets can then be sent to a proxy server without requiring clients to reconfigure their applications [Allen 01].

Non-transparent proxies are a bit more complex. The client system needs to be configured to specifically forward all traffic to the proxy at a single target port. For example, the SOCKS protocol is typically used to tunnel all traffic from the client to the proxy using a target port of TCP/1080 or TCP/8080. This can be done by using only proxy-aware applications or by installing special client software.

The disadvantages of non-transparent proxies occur when there are a large number of hosts that need to be maintained with the proxy client software. Often proxy client software is only provided for Windows machines. And using such software can be a problem for users that use their laptop to connect to your local network during the day and remotely via an ISP during off hours or while traveling. If a proxy product claims to support SOCKS, it is not a transparent proxy. However, non-transparent proxies can provide a grater level of security by hiding routing entries, and providing for proxy-specific user authentication and data encryption between the client and the proxy server [Brenton 01], [Brenton 99].



Networked Systems Survivability

Network Address Translation (NAT)

Translates between public IP addresses and internal network IP addresses, for both incoming and outgoing traffic

Hides internal network TCP/IP addresses and server/workstation configuration data

Solves the problem of scarcity of IP addresses

Has several variations:

- Static NAT (one to one)
- Dynamic NAT (many to one)
- Port Address Translation (PAT)

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 23

NAT functions very much like a Private Branch Exchange (PBX) in a telephone system. A company telephone system may have several hundreds of telephone extensions within the company. Each telephone has its own internal "extension" number it uses to call others in the company. When it calls someone outside the company, the outside sees the number of the "trunk" line the PBX system uses and not the extension number of the user’s telephone. The actual connection between the outside trunk and the inside user is maintained temporarily by the telephone system [Smith 01].

NAT can be configured to allow insiders to get out without allowing outsiders to get in. When a request is sent through the firewall, the NAT application substitutes its own address for the source address field. When a reply comes back to the NAT application, it replaces its own address in the destination field with that of the original client making the request. In this case, external hosts cannot find the internal host addresses because they are aware of only one IP address, the firewall [Ogletree 01], [Smith 01].

However, NAT can also be configured to offer a connection to an inside host from the outside world by redirecting a request to the NAT firewall to a specific socket of a service hosted on the inside (referred to as port forwarding; http://www.e-infomax.com/ipmasq/ and see also page 31).

NAT solves the problem of the scarcity of IP addresses. The administrator of an internal network can choose reserved IP addresses, e.g., 10.x.x.x range or 192.168.x.x range. These addresses do not have to be registered with an authority and can be used however the administrator wants. Thus, a site with few or one Internet IP address can have hundreds of computers each with their own IP address without denying any of its users Internet access [Ogletree 01].

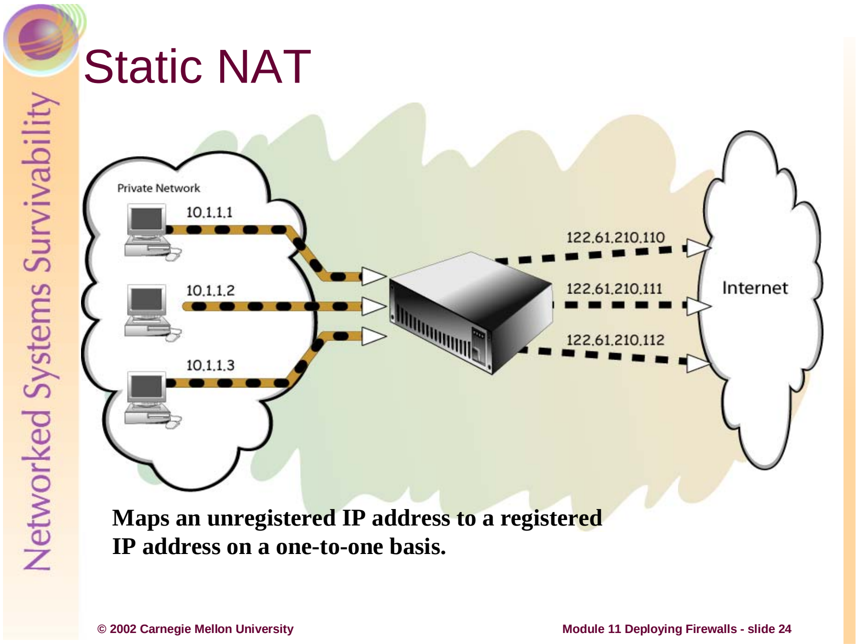
Each outbound connection’s source address is remapped so it appears to have originated from the firewall’s external address. This means that it is not possible to directly access any internal addresses on your network. NAT changes the destination address for inbound packets from the (public, visible) firewall address to the (private, invisible) internal host address. NAT also converts client-side TCP and UDP port numbers to prevent port number conflicts as a result of the address translation (sometimes referred to as port-level NAT or PAT). Using NAT reduces address space storage and eliminates the need to renumber when changing your ISP. NAT contains a pool of available global addresses that are constantly reused [CBT 98].

NAT variations are described on the following pages.

For further information, refer to:

[Egevang 94] Egevang, K. Francis, P. *RFC 1631 The IP Network Address Translator (NAT)*. Internet Engineering Task Force Network Working Group, 1994. Online: http://www.ietf.org/rfc/rfc1631.txt

Tyson, Jeff. “How Network Address Translation Works.” Online: http://www.howstuffworks/nat.htm and Cisco Systems Inc. “How NAT Works.” Online: http://www.cisco.com/warp/public/556/nat-cisco.shtml



Networked Systems Survivability

Static NAT

Maps an unregistered IP address to a registered IP address on a one-to-one basis.

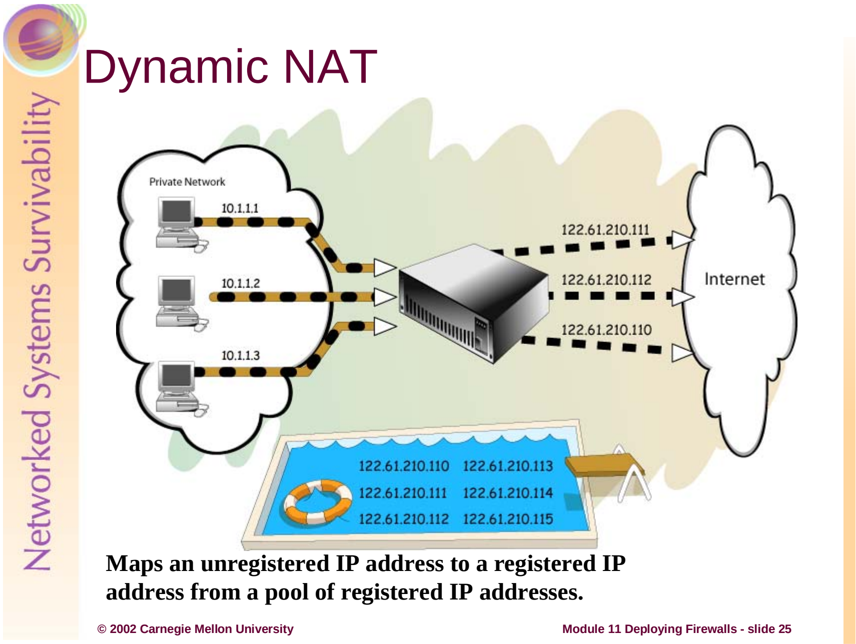
© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 24

NAT can work in several ways:

Static NAT is used to map individual internal private IP addresses to a unique legal IP address. This is useful when you have one or two systems on a private address network that need to be accessed by hosts on the Internet. Since there is a direct one-to-one IP mapping, the firewall does not need to change the port number like it does with overloading/port address translation (see later slide). This can be useful for services that attempt to assign a fixed source port as well as a fixed target port [Brenton 01].

[Static NAT, Dynamic NAT, and Overloading graphics are taken from [Tyson/Cisco].]



Networked Systems Survivability

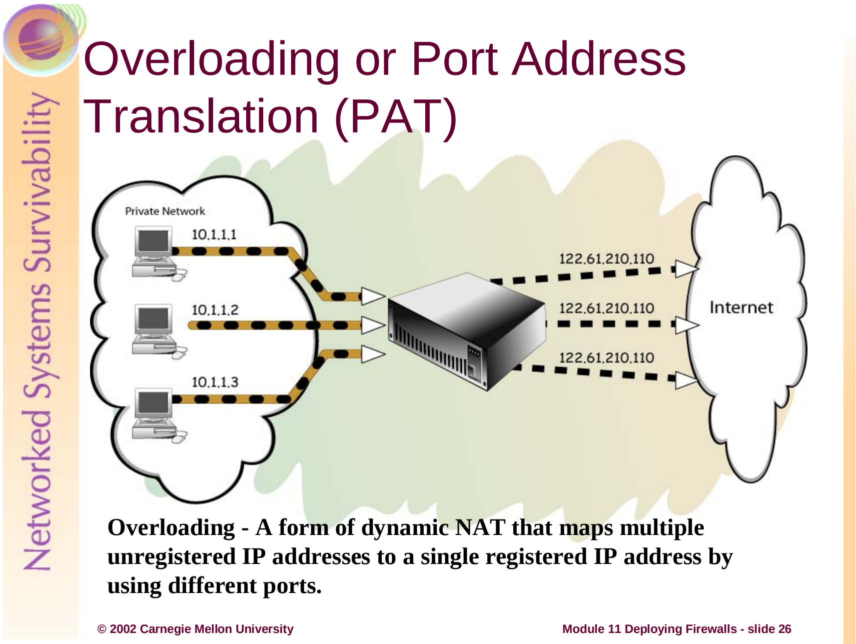
Dynamic NAT

Maps an unregistered IP address to a registered IP address from a pool of registered IP addresses.

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 25

Dynamic NAT is used to provide users who have been assigned a private IP address a temporary legal IP address when accessing Internet resources. The method is similar to static NAT, except each private address is not mapped to one specific legal address. Rather, users are allocated a legal address on the fly when they attempt to access the Internet. When they are done, the address goes back into the pool and can be used by another user. If all pool addresses are in use and an additional user attempts to access resources, most NAT devices will revert to overloading/port address translation (see next slide) [Brenton 01].



Networked Systems Survivability

Overloading or Port Address Translation (PAT)

Overloading - A form of dynamic NAT that maps multiple unregistered IP addresses to a single registered IP address by using different ports.

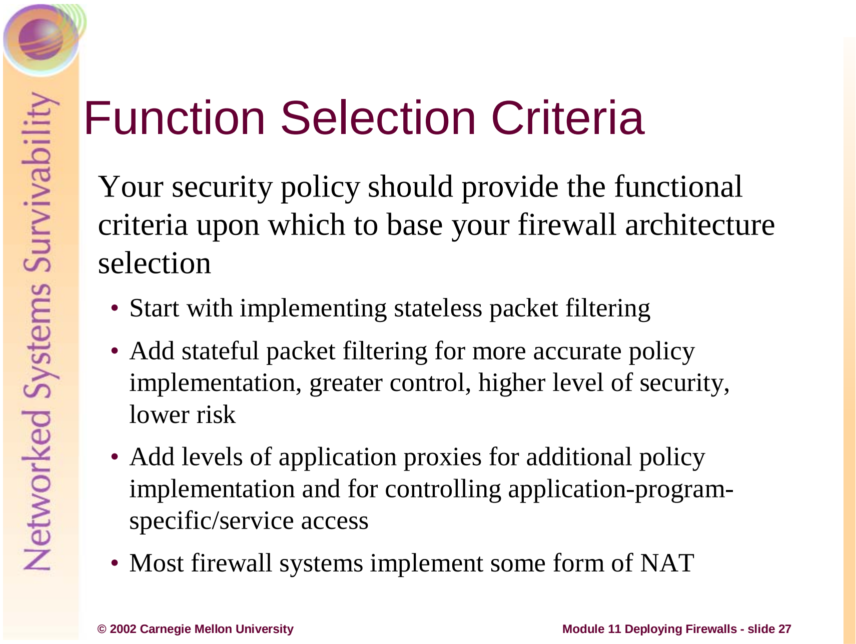
© 2002 Carnegie Mellon University Module 11 Deploying Firewalls - slide 26

Overloading allows multiple internal clients to access the Internet, all appearing to originate from the same IP address. All outbound traffic has the source IP address changed so that it appears to originate from the external IP address of the firewall. The firewall also changes the source port number and uses this as an accounting method to keep track of multiple sessions [Brenton 01].

Overloading converts client-side TCP and UDP port numbers to prevent port number conflicts as a result of the address translation. It is also known as PAT (Port Address Translation), single address NAT, port-level multiplexed NAT, or Network Address Port Translation (NAPT) [Tyson, Cisco].

For example, CGI scripts and Java applets that access a database server, or some other type of server, are becoming more common as businesses do more e-commerce. This presents a security problem because the script/applet must be outside the firewall where it can be accessed from the Internet but the database containing sensitive customer data is behind the firewall where it is more difficult to attack. Enter PAT to the rescue. On the firewall, IP packets coming into a specific port number are re-written and forwarded to the internal server providing the requested service. The reply packets from the server are re-written to make it appear as if they originated on the firewall. Thus, PAT can be used to secure internal servers for external access [Ogletree 01].

If you need to limit the number of outbound connections, static NAT is likely the best choice. If you have a large number of clients that need to establish connections simultaneously and you do not have a sufficient number of valid IP addresses for NAT use, then dynamic may be the way to go [Ogletree 01].


Networked Systems Survivability

Function Selection Criteria

Your security policy should provide the functional criteria upon which to base your firewall architecture selection

- Start with implementing stateless packet filtering
- Add stateful packet filtering for more accurate policy implementation, greater control, higher level of security, lower risk
- Add levels of application proxies for additional policy implementation and for controlling application-program-specific/service access
- Most firewall systems implement some form of NAT

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 27

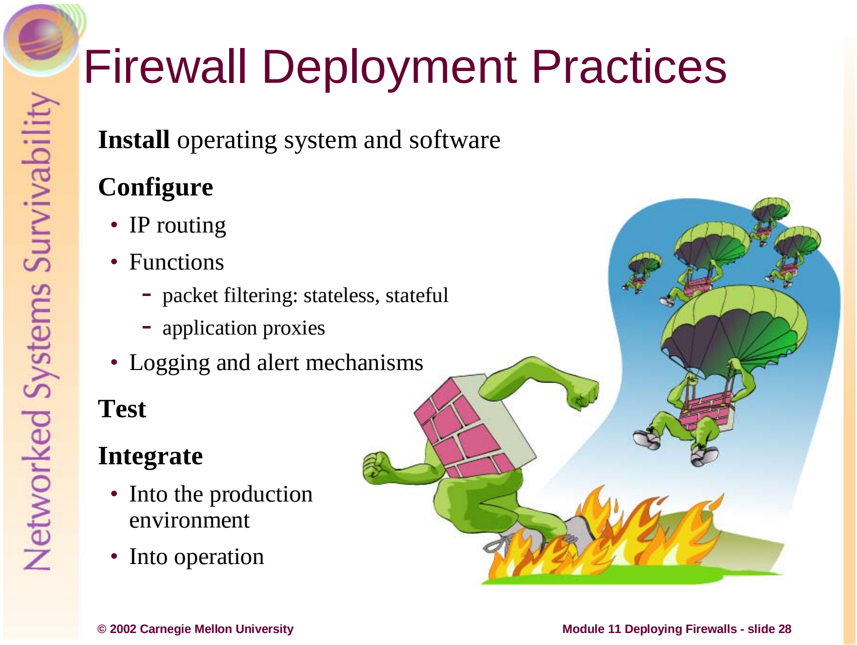
Firewall functions operate at different OSI protocol layers using different criteria to pass or restrict traffic. The lowest layer at which a firewall can operate is layer 3 (network). This layer is concerned with the routing of packets to their destination. At this layer, a firewall can determine if a packet is from a permitted source but cannot be concerned with the contents of the packet. Firewall functions that operate at the next layer, the transport layer, are able to make more sophisticated decisions about accepting or denying packets because they know more about a packet. At the application layer, firewall functions have even more information available and can use even more stringent criteria to permit or deny packets. Lower layer filtering is faster; higher layer filtering provides for greater examination but is slower [Smith 01].

Stateless packet filtering should be sufficient to implement some portion of your organization's network security policy. If the entire policy can be implemented with stateless packet filters, then other firewall functions may not be required. If some elements of your policy cannot be implemented, then you need to consider stateful packet filters and proxies [Allen 01].

Stateful filtering can protect against crafted, spoofed packets (such as ones with the ACK bit set) whereas stateless filtering cannot [Brenton 01].

The biggest limitation of a stateful packet filter is that it cannot make filtering decisions based upon payload which is the actual data contained within the packet. A proxy handles traffic on a per port basis and its real strength is in screening data content. [Brenton 99, 01] Application proxies can screen data streams for potential content-based threats, e.g., sendmail attacks, Java, ActiveX, or other code riding on top of HTTP. They can also process and manipulate packet header and content information [Smith 01].

When using a packet filter (stateless or stateful), IP forwarding needs to be enabled. This is because the firewall software only filters the traffic. It cannot transmit it out the opposite interface. The operating system handles this functionality. When deploying a proxy however, disable IP forwarding as the proxy takes care of packet delivery. This is one of the reasons why proxy technology is considered more secure. If the packet filter fails, the operating system might forward packets unchecked. If the proxy fails, there is no longer a path across the system [Brenton 01].



Networked Systems Survivability

Firewall Deployment Practices

Install operating system and software

Configure

- IP routing
- Functions
  - packet filtering: stateless, stateful
  - application proxies
- Logging and alert mechanisms

Test

Integrate

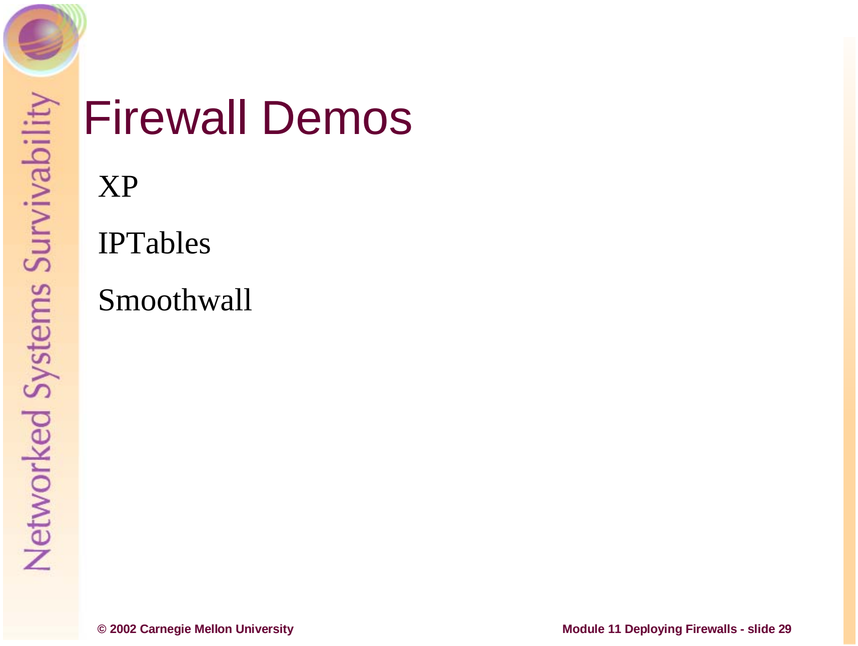
- Into the production environment
- Into operation

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 28

This section describes four practice categories and seven practices that aid in the effective and secure deployment of firewalls, minimizing disruption to ongoing operations in the case of installing a replacement firewall.

Further details for all deployment practices can be found in [Allen 01], Chapter 4.



Networked Systems Survivability

Firewall Demos

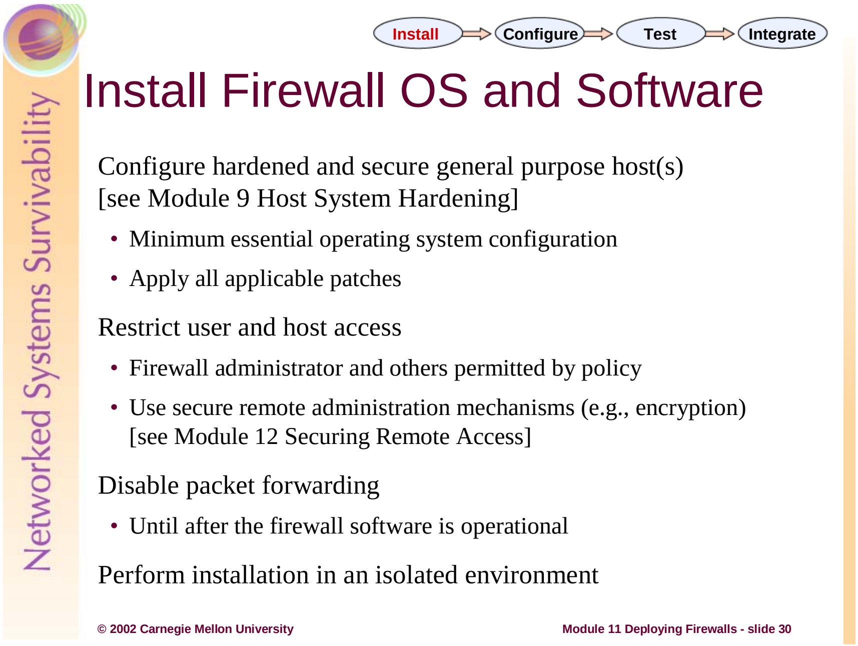
XP

IPTables

Smoothwall

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 29



Install
Configure
Test
Integrate
Networked Systems Survivability

Install Firewall OS and Software

Configure hardened and secure general purpose host(s)
[see Module 9 Host System Hardening]

- Minimum essential operating system configuration
- Apply all applicable patches

Restrict user and host access

- Firewall administrator and others permitted by policy
- Use secure remote administration mechanisms (e.g., encryption)
[see Module 12 Securing Remote Access]

Disable packet forwarding

- Until after the firewall software is operational

Perform installation in an isolated environment

© 2002 Carnegie Mellon University   Module 11 Deploying Firewalls - slide 30

Configure hardened and secure general purpose host(s)

First you need to install and configure the operating system that will execute the firewall software followed by installing and configuring the firewall software. These two steps should be performed on the firewall hardware you intend to use in your production environment but deployed in the test environment and configuration (Refer to “Test the Firewall System” for information on using a test configuration). You need to ensure that all hardware and software are properly configured and operate as expected to the extent possible in the test configuration.

You need to configure the operating system on your firewall host in the minimum essential configuration so that only those services necessary for firewall operation and maintenance are included. You need to include all applicable patches or fixes for both the operating system and the firewall software.

Keep in mind that packet filtering functions typically run in the operating system kernel (for performance reasons) and, therefore, packet filtering software is fairly sensitive to a specific kernel version and release number.

Once you are satisfied that the operating system and the firewall software is successfully installed, you should repeat the sequence to ensure that the process can be done again. The second time, document it. The third time, have an outside person who was not involved in the first two installations follow the documentation to see if it is correct and complete.

Take appropriate steps to ensure that any redundant systems are in a state consistent with the systems to be used in production. Ensure that you can easily switch between your primary firewall system and any redundant systems.

Your installed environment may not have all of the necessary troubleshooting and support tools necessary to determine what has happened if anything goes wrong during the installation process. You may need to install the firewall system on another host that has better diagnostic tools if you run into problems. After you understand the problems and know how to compensate for them, you can complete the installation on the production hardware.

Restrict user and host access

The only users who should have access to your firewall system are the firewall system administrator, those authorized by policy, and individuals involved in operating and maintaining your information technology infrastructure.

For some firewall products, the process of installing the firewall software will automatically disable access to the firewall system by all users (except those mentioned above) if you have not already disabled their access before installation.

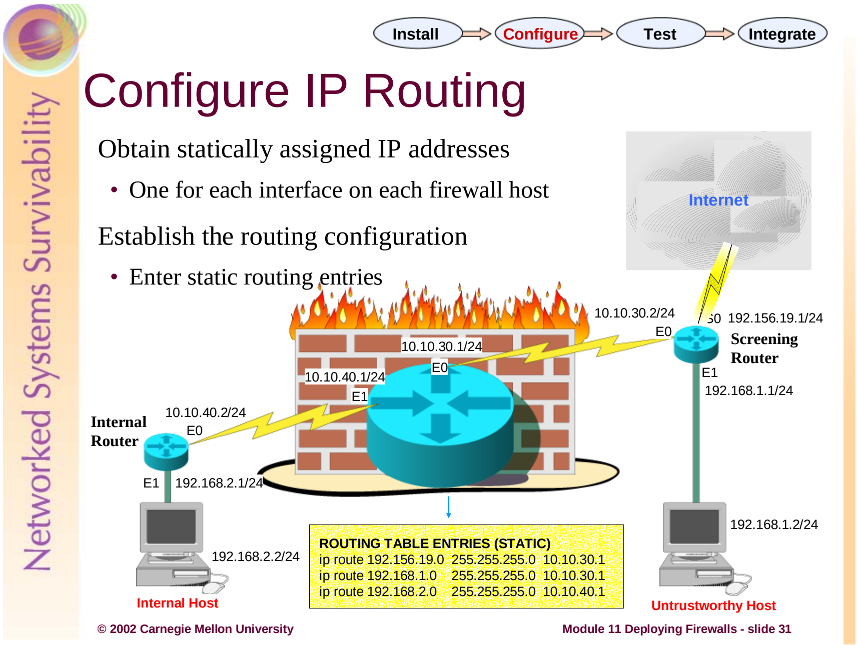
We recommend that you allow remote access to your firewall system only via mechanisms that are strongly authenticated and strongly encrypted, even on your organization's internal networks. Some firewall products provide the capability to restrict the administrative client to a specific IP address and a specific port. We do not believe that this is adequate security; encryption providing mutual authentication is required as well. IP addresses and ports are too easily spoofed.

Disable packet forwarding

Make sure packet forwarding is disabled until after the firewall software is operational. While booting firewall hosts, there may be an interval of time after the operating system is functional, including networking, but the firewall software is not yet functional. During this interval, packets may flow freely through the firewall system. Make sure that no packets are forwarded before the firewall software is functioning by doing one or both of the following:

- disable IP routing before any interfaces are enabled
- do not enable network interfaces before the firewall software is functional

In addition, when installation is complete, perform a backup of the entire firewall system. Use this backup to restore the production system (or one identical to the test firewall system) for operation. Verify that both the operating system and the firewall software operate properly from the restored backup version [Allen 01].



Configure IP Routing

Obtain statically assigned IP addresses

- One for each interface on each firewall host

Establish the routing configuration

- Enter static routing entries

ROUTING TABLE ENTRIES (STATIC)

```
ip route 192.156.19.0  255.255.255.0  10.10.30.1
ip route 192.168.1.0   255.255.255.0  10.10.30.1
ip route 192.168.2.0   255.255.255.0  10.10.40.1
```

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 31

Obtain statically assigned IP addresses

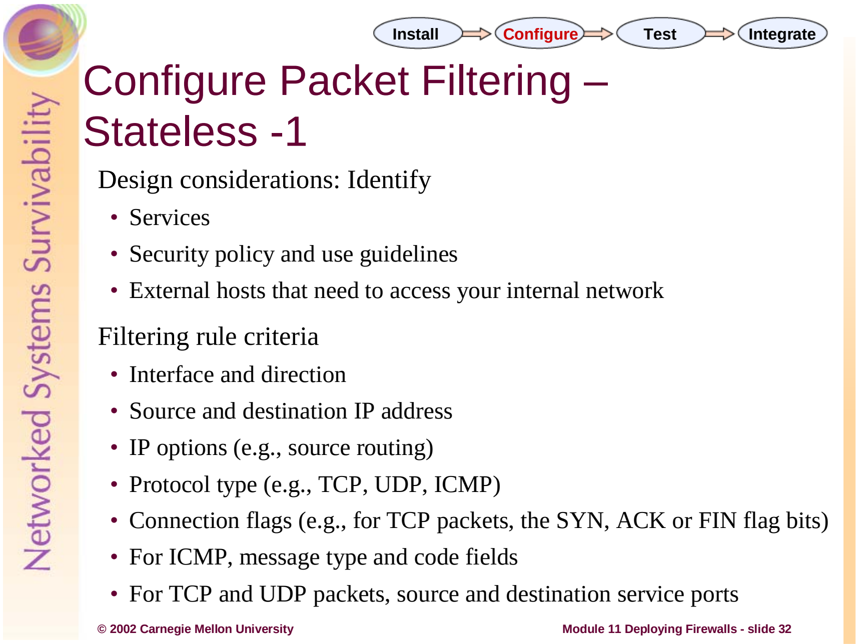
Each network to which a firewall system is attached has a procedure to obtain new IP addresses. For the Internet, this is obtained from the Internet Service Provider (ISP) that will connect to your firewall. For internal networks, including any DMZ networks you intend to establish, you must obtain IP addresses from within your organization.

Establish the routing configuration

A firewall system's routing configuration reflects its view of the topological configuration of the networks to which it is attached. Most firewall systems’ routing configurations rarely change; they are typically static. Dynamic updates to routing configurations do occur. However, the large majority of firewalls today have two interfaces — one to the Internet and one to the organization’s internal network. In this case, routing is static. Most ISPs handle all dynamic routing, presenting a static interface to their customers’ systems at all times.

If you are replacing an existing firewall system or router, thoroughly examine the routing configuration of your system to determine the network topology that it describes. Ensure, as a first step, that the routing configuration of the new firewall system is consistent with your current system before departing from that configuration [Allen 01]. For further information on routing protocols and the process of establishing a routing configuration, refer to [Comer 95].

In the above slide, static routes have been implemented on the firewall to provide connectivity to the internal network, the Internet, and a DMZ network. The firewall will test packets against its rule set, and then forward permitted packets using its routing table.



Networked Systems Survivability

Install → Configure → Test → Integrate

Configure Packet Filtering – Stateless -1

Design considerations: Identify

- Services
- Security policy and use guidelines
- External hosts that need to access your internal network

Filtering rule criteria

- Interface and direction
- Source and destination IP address
- IP options (e.g., source routing)
- Protocol type (e.g., TCP, UDP, ICMP)
- Connection flags (e.g., for TCP packets, the SYN, ACK or FIN flag bits)
- For ICMP, message type and code fields
- For TCP and UDP packets, source and destination service ports

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 32

Design Considerations

Each service (such as email, public and private web access, domain name services, file transfer) that is offered and requested operates with unique rules and differing protocols. Service operating requirements need to be considered in firewall design. If your firewall permits services such as Telnet or FTP, your firewall design needs to consider some form of strong authentication and/or the use of proxies [Ogletree 01].

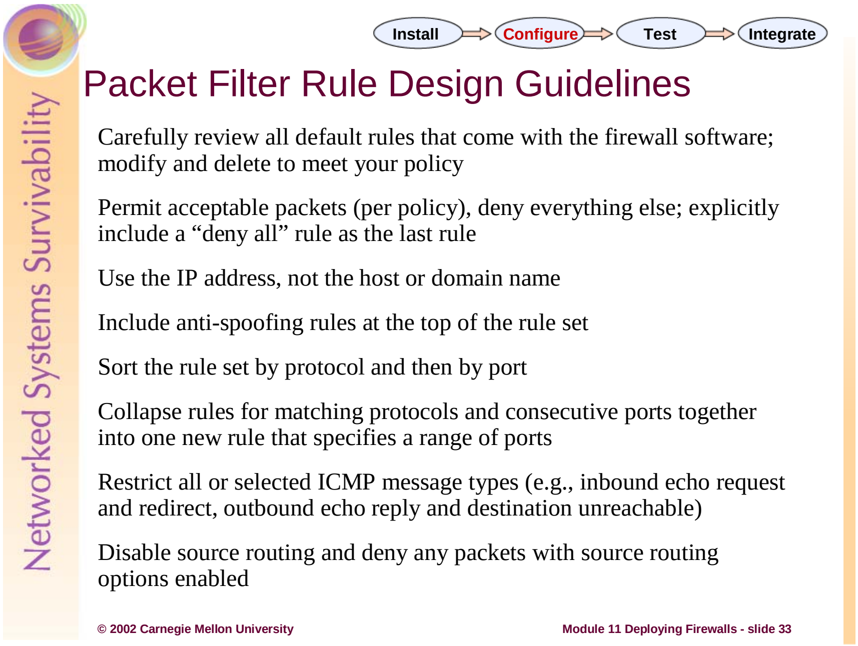
Your policy may require you to limit what users can publicly access via the Internet and monitor and enforce such limitations. Your firewall may also need to block access to certain services and sites.

Your firewall design needs to take into account access through the firewall to publicly-offered services as well as access to internal hosts by trusted parties and the use of secure remote administration services by authorized administrators working remotely.

Filtering rule criteria

As discussed in Module 5 (TCP/IP Security), each protocol behaves in a prescribed manner and contains well-defined header contents. Protocol behavior and packet header fields can generally be used to specify packet filtering rules. Filter rules can indicate permit or deny actions based on protocol and such header fields as source and destination address, port, and flag bits.

The criteria used in each filtering rule for determining the disposition can be arbitrarily complex. For a router with packet filtering, there may be multiple points in the routing process where the rules are applied; typically, for arriving packets, they are applied at the time a packet is received and, for departing packets, they are applied immediately before a packet is transmitted. There may be different rule sets at each point where filtering is applied [Allen 01].



Install → Configure → Test → Integrate

Networked Systems Survivability

Packet Filter Rule Design Guidelines

Carefully review all default rules that come with the firewall software; modify and delete to meet your policy

Permit acceptable packets (per policy), deny everything else; explicitly include a “deny all” rule as the last rule

Use the IP address, not the host or domain name

Include anti-spoofing rules at the top of the rule set

Sort the rule set by protocol and then by port

Collapse rules for matching protocols and consecutive ports together into one new rule that specifies a range of ports

Restrict all or selected ICMP message types (e.g., inbound echo request and redirect, outbound echo reply and destination unreachable)

Disable source routing and deny any packets with source routing options enabled

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 33

Carefully review all default rules that come with the firewall software; modify and delete to meet your policy

Vendors often create firewall filtering configurations with ease of use in mind more than security. Some of the filtering rules are not obvious and will likely not comply with your security policy.

Permit acceptable packets (per policy), deny everything else; explicitly include a “deny all” rule as the last rule

You can generally assume that the last rule in every rule set of every firewall system is to deny all packets. However, we recommend that you explicitly add this rule to remind you that this is the policy you are implementing and to express the rule set more completely.

Use the IP address, not the host or domain name

Host and domain names are easier to spoof than IP addresses. If a host’s address changes, the firewall will not recognize the change until the filter configuration is reloaded and the names are converted to addresses again.

Include anti-spoofing rules at the top of the rule set

You want your firewall to block spoofed packets as the first order of business to ensure that some subsequent rule does not inadvertently permit their entrance. In addition, for anti-spoofing rules to work as intended, your firewall must be able to distinguish between arrival and departure on each interface independently. You need to specify sets of rules that reference interfaces and direction; otherwise, you cannot implement anti-spoofing rules without interfering with other rules. To defend against spoofing attacks, block any packet that comes from outside and that has an inside source IP address.

Reject all packets entering through the external network adapter that have source addresses that can only originate from inside the protected network. These packets are probably from someone trying to use address spoofing [Ogletree 01].

Refer to RFC 2267, *Network Ingress Filtering: Denial of Service Attacks which employ IP Source Address Spoofing*, available at http://www.ietf.org/rfc/rfc2267.txt, for more information about spoofing and designing anti-spoofing rules.

Sort the rule set by protocol and then by port

This aids understanding and legibility.

Collapse rules for matching protocols and consecutive ports together into one new rule that specifies a range of ports

This helps simplify the rule set and aids in legibility.

Restrict selected ICMP message types (e.g., inbound echo request and redirect, outbound echo reply and destination unreachable)

Block packets for any protocol that is not used on your network or that can allow someone outside of your network to reconfigure how your network operates. For example, the ICMP redirect message type tell your routers that a destination is not reachable and to reconfigure its tables to change the route to a particular network, something you may want to disallow. Instead of inbound echo request and outbound echo reply messages, allow your users to use PING to test connectivity to outside hosts, but not the other way around. Blocking outbound destination unreachable and service unavailable messages, makes is more difficult for an intruder to determine what is not reachable or what services are not offered.

Disable source routing and deny any packets with source routing options enabledSource routing is a function of IP routing that allows the packet originator to influence routing decisions as the packet traverses the network. We recommend that you disable all source-routing functions in your firewall's router and, if possible, deny any packets that have specified source-routing options.

Additional guidelines

The order in which the rules are applied is important; they are generally applied in a top- down order. If they are stored in the wrong order, wrong packets could end up being permitted and valid packets could be discarded. Rules are usually ordered from most restrictive to least restrictive [CBT, Smith 01].

If your firewall has separate rule sets for arrival and departure on each interface, repeat the rules in the arrival rule set of each interface and in the departure interface of the others. This reduces the possibility of an oversight.

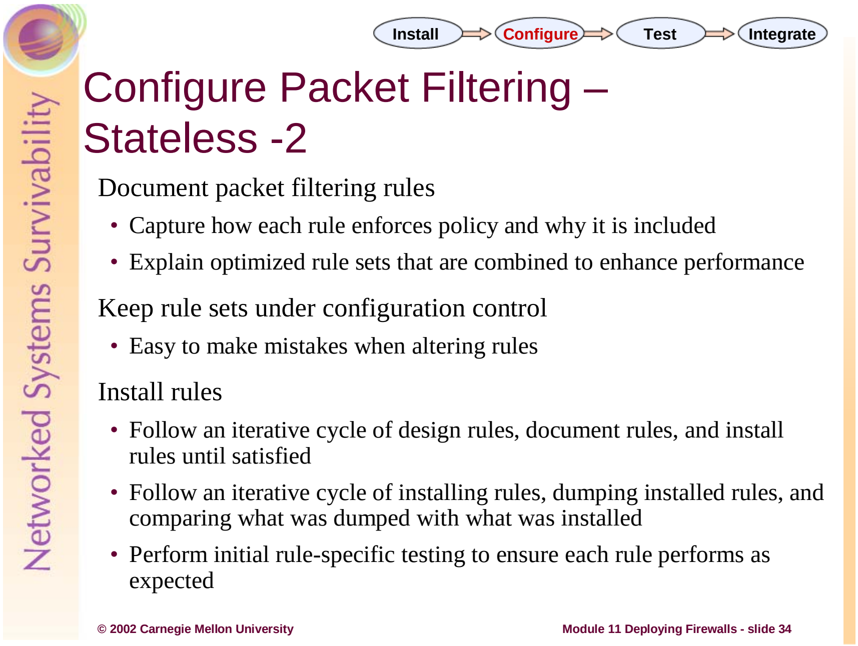
IP implementation flaws can compromise packet filtering effectiveness. Such flaws include fragmentation reassembly errors and incorrect interpretation of invalid packet headers.

Consider adding redundant filter rules (one that either permits or denies packets that are already permitted or denied by another rule) just so that you can specify different logging options for that rule. If you do this, make sure to document the fact that the rule is redundant and not essential to the implementation of your security policy.

UDP (User Datagram Protocol) and ICMP (Internet Control Message Protocol) are connectionless protocols, i.e., they have no concept of an "established" state such as TCP and each message stands on its own with no relationship to past and future messages. As a result, there is no completely safe way to allow "return" UDP and ICMP packets with traditional packet filtering – they simply cannot be associated with an outgoing packet. However, a number of critical services (such as DNS using UDP port 53) depend on these types of packets. If possible, use stateful filtering techniques features to filter connections based on state [Allen 01].

Applying the same rules to an entire network, rather than slightly different rules to individual hosts on that network, minimizes the performance cost of filtering [CBT98].

Refer to Module 5 and [Stevens 94] for details on TCP, UDP, and ICMP.


Networked Systems Survivability
Install
Configure
Test
Integrate

Configure Packet Filtering – Stateless -2

Document packet filtering rules

- Capture how each rule enforces policy and why it is included
- Explain optimized rule sets that are combined to enhance performance

Keep rule sets under configuration control

- Easy to make mistakes when altering rules

Install rules

- Follow an iterative cycle of design rules, document rules, and install rules until satisfied
- Follow an iterative cycle of installing rules, dumping installed rules, and comparing what was dumped with what was installed
- Perform initial rule-specific testing to ensure each rule performs as expected

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 34

Document packet filtering rules so that anyone reading the documentation and configuration data is able to see how each filtering rule enforces its part of the firewall policy. Describe why each rule is included.

Specifying the detailed rules that implement your firewall policy can be a very complex and time-consuming process. It is very common that several rules are required to achieve one integral function. Group the rules that go together and include comments or other documentation that explains what the group does (to the extent that the filter syntax permits).

Most commercial firewall products provide complex configuration managers and user interfaces for specifying rules. Some products provide the capability to aggregate related sets of rules into groups. However, the language used to express the rules is typically network-based, not policy-based. This means that you can specify a rule and understand what it does from a network perspective, but you may still have little idea of the policy implications.

Establishing and maintaining your understanding of the rules from both a policy viewpoint and a network viewpoint at the same time is very difficult. Often firewall system performance deteriorates as the number of filtering rules increase. An administrator may be forced to combine rules to optimize performance and needs to describe what the optimized rule set does and how it performs. [Allen 01]

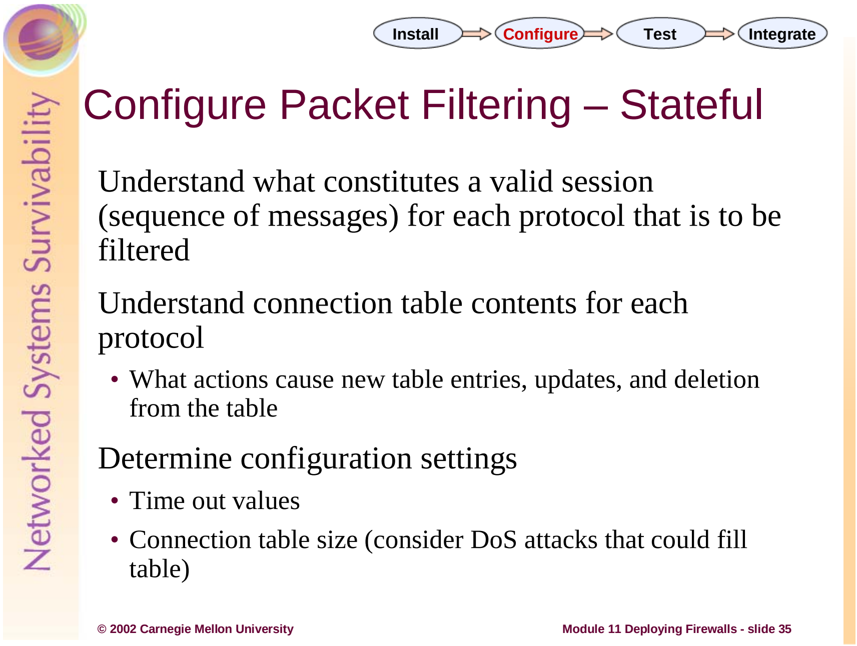
Keep rule sets under configuration control

It is easy to make mistakes in altering rules as requirements change. We recommend keeping the rule sets under configuration control so that you can look back to prior configurations when necessary (and it will be necessary).

Install rules

Now that you have one or more sets of rules, install them in your firewall test environment. Most firewall software has a mode of operation that allows the installed filters to be dumped to a file for examination. Install your filters, dump them, and compare the two. Sometimes you will find ambiguities in the input language that result in your implementing something different from what you intended. Continue this install-dump-compare cycle until you are satisfied that you have installed what you intended to install.

In actual use, all three steps of this practice (design, document, install) are performed in parallel using a process of successive refinement. It is also common to perform initial rule-specific testing during this activity to ensure that each rule does what you expect it to do [Allen 01].


Install
Configure
Test
Integrate
Networked Systems Survivability

Configure Packet Filtering – Stateful

Understand what constitutes a valid session (sequence of messages) for each protocol that is to be filtered

Understand connection table contents for each protocol

- What actions cause new table entries, updates, and deletion from the table

Determine configuration settings

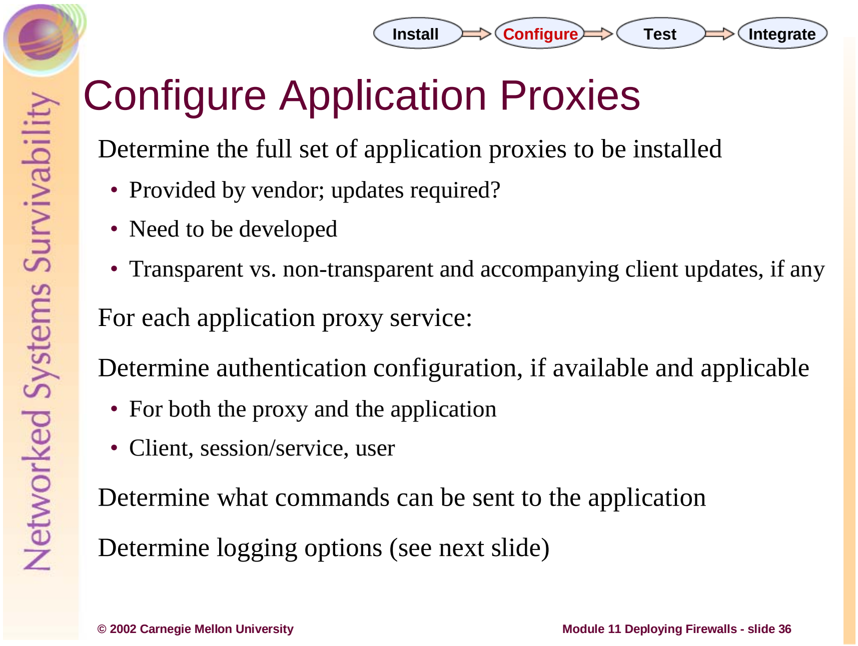
- Time out values
- Connection table size (consider DoS attacks that could fill table)

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 35

By remembering the state of a connection, a stateful filter can make intelligent decisions as to which packets are true reply traffic (based on a previously accepted request) and which packets are new. A valid reply matched with an existing connection table entry will result in the reply passing through.

Every time a new packet is sent to the firewall and is successfully matched with a packet filter rule, the firewall makes a new connection table entry recording the start of a new session. A connection table entry typically contains the protocol, the time remaining before the connection table entry is purged, last state information, source IP address and port, and destination IP address and port. Some firewalls also support recording and testing sequence numbers and current flag settings. A connection table entry is removed when the session ends (RST or FIN packet exchange) or times out based on a configuration setting. A connection table may be limited in terms of the number of entries it can contain [Brenton 01].


Networked Systems Survivability

Install → Configure → Test → Integrate

Configure Application Proxies

Determine the full set of application proxies to be installed

- Provided by vendor; updates required?
- Need to be developed
- Transparent vs. non-transparent and accompanying client updates, if any

For each application proxy service:

Determine authentication configuration, if available and applicable

- For both the proxy and the application
- Client, session/service, user

Determine what commands can be sent to the application

Determine logging options (see next slide)

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 36

You need to install proxy service code on the firewall for each desired application, and determine whether or not the vendor has provided a proxy, and whether the proxy is transparent or non-transparent [CBT 98].

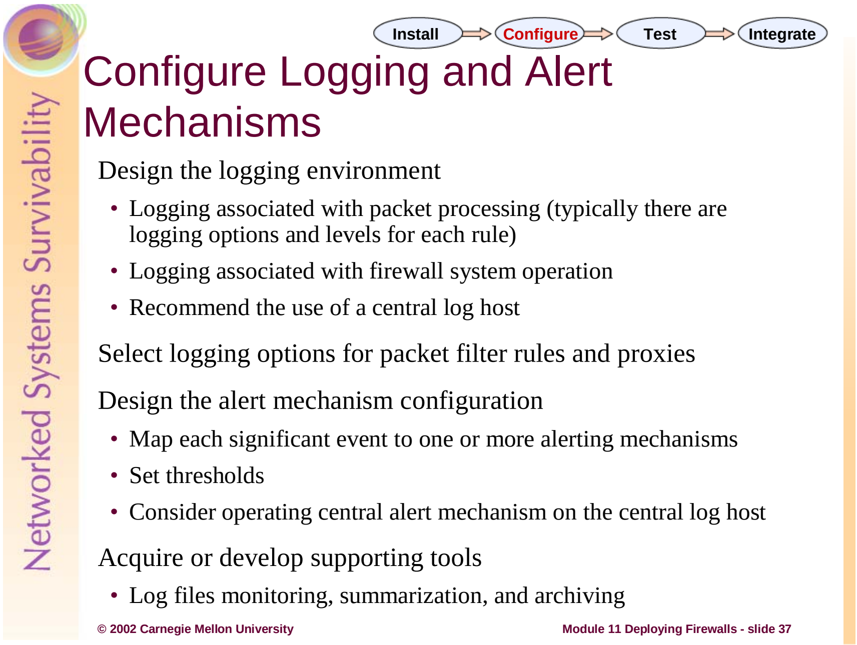
With respect to authentication and access control [CBT 98]:

- Proxy services can be configured to require additional user authentication. Access is permitted to specific host systems only.
- A proxy can limit the subset of commands that are sent to the application.
- Each proxy service can be configured to require its own authentication.
- Stateful packet filtering can be used in concert with application proxies. For example, an authenticated user who has used a particular service before (resulting in a connection table entry) can access the application but must use the same service as before.

Each proxy logs detailed audit information recording all traffic, each connection, and connection duration.

Other than reading its initial configuration file, a proxy does not generally permit disk access.

Application proxy filtering rules are much easier to configure and test than packet filtering rules [CBT 98].



Install → Configure → Test → Integrate

Networked Systems Survivability

Configure Logging and Alert Mechanisms

Design the logging environment

- Logging associated with packet processing (typically there are logging options and levels for each rule)
- Logging associated with firewall system operation
- Recommend the use of a central log host

Select logging options for packet filter rules and proxies

Design the alert mechanism configuration

- Map each significant event to one or more alerting mechanisms
- Set thresholds
- Consider operating central alert mechanism on the central log host

Acquire or develop supporting tools

- Log files monitoring, summarization, and archiving

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 37

The most important reason for logging is to ensure the continued operation of the firewall system. Logged events related to the operational status of the firewall are critically important in preventing and recovering from failures. Logs are also an important auditing tool to ensure that the proper security configurations (e.g., packet filters, proxies) are installed on the firewall system. Logs of this sort are generally small and can have long-term value for a variety of purposes.

Completely apart from firewall operational concerns are concerns about security. Logs can be useful in determining how an intrusion might have occurred for the purpose of improving the quality of the firewall implementation. Logs for this purpose have value only over the time period where intrusions can be reasonably resolved — in our experience no more than three to four weeks.

Logs might also be used in intrusion detection. Intrusion detection is the process of detecting attempted, failed, and successful attacks against your network. Logs for this purpose may have historical value. Refer to [Escamilla 98] and [Allen 01] for more information on using logs for the purpose of intrusion detection.

Since log files are typically voluminous and difficult for humans to process, you should use alert mechanisms to notify you of any significant event. It is generally impractical to depend on manual analysis of logs to detect significant events.

Design the logging environment

Determine:

- Log files locations
- Expected sizes of log files
- Logging rates
- Level of access to log files and by whom
- Whether or not to encrypt log files
- Log files backup and recovery approach
- Approach for handling archived log information including how long it should be retained and when it should be discarded

Select logging options for packet filter rules and application proxies

Consider logging on conditions such as:

- Packet denied upon arrival at the firewall system
- Packet denied upon departure from the firewall system
- Packet arrived or departed within a specified time or date interval

Address how best to summarize individual filter logs. These can be useful for seeing trends, and are generally more worthwhile to retain for an extended period than individual log entries.

Determine under what conditions you might want to track a particular kind of traffic. Consider adding a redundant filter rule to specify different logging options for that rule.

Design the alert mechanism configuration

Significant events may include:

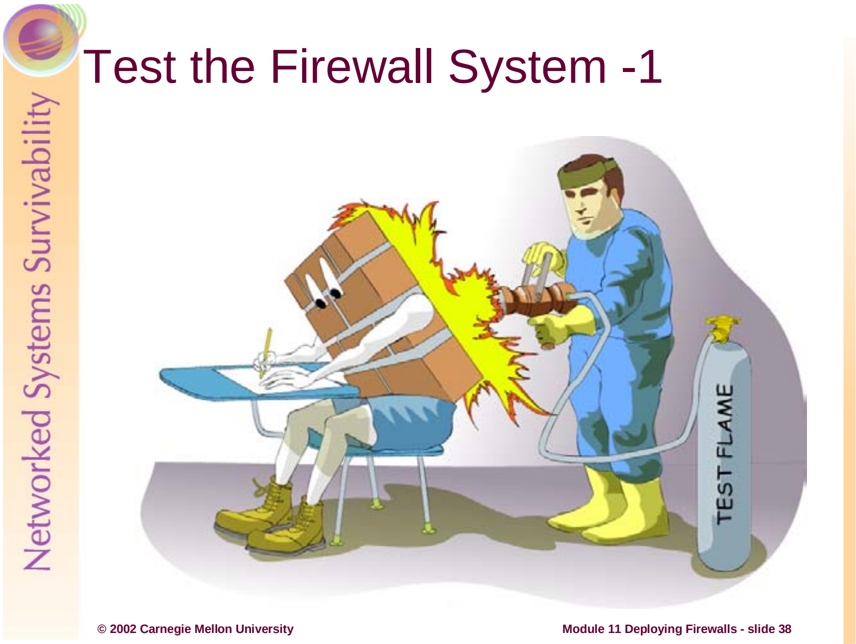
- Unsuccessful user and host login attempts
- Modification or disabling of packet filters in the firewall system
- Successful logins to the firewall system
- Changes to certain files on the firewall system
- Operational events such as logs full, memory or disk shortages, system reboots

If you have chosen to implement a central log host, consider establishing a central alert mechanism that also operates on the central log host.

Alerting mechanisms may include:

- Email displayed in a high priority window on the administrator's workstation
- Phone/voice mail
- Pager notification

[Allen 01]



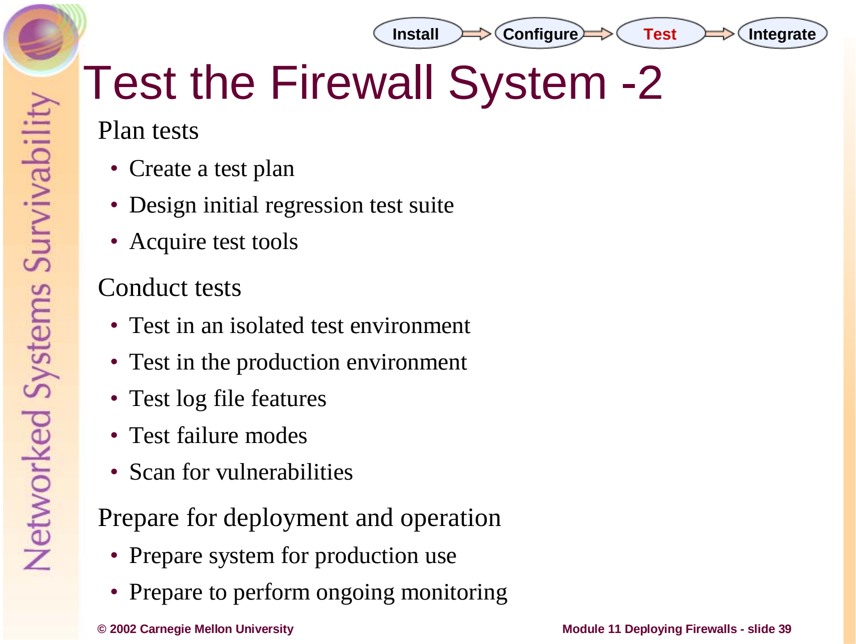
Test the Firewall System -1

Networked Systems Survivability

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 38

Refer to the next page for a description of firewall testing practices.


Install
Configure
Test
Integrate
Networked Systems Survivability

Test the Firewall System -2

Plan tests

- Create a test plan
- Design initial regression test suite
- Acquire test tools

Conduct tests

- Test in an isolated test environment
- Test in the production environment
- Test log file features
- Test failure modes
- Scan for vulnerabilities

Prepare for deployment and operation

- Prepare system for production use
- Prepare to perform ongoing monitoring

© 2002 Carnegie Mellon University Module 11 Deploying Firewalls - slide 39

The purpose of the test activity is to verify that the firewall system works as intended. You should plan testing activities to demonstrate that routing, packet filtering, and logging and alert capabilities perform as designed; test recovery plans for firewall system failures; and design your initial regression testing suite.

The features that should be tested include:

- hardware (processor, disk, memory, network interfaces, etc.)
- operating system software (booting, console access, etc.)
- firewall software
- network interconnection equipment (cables, switches, hubs, etc.)
- firewall configuration software
  - routing rules
  - packet filtering rules and associated logging and alert options

Plan Tests

<u>Create a test plan</u>

The most common cause of firewall security breaches is a misconfiguration of the firewall system. Knowing this, make thorough configuration testing – both of the firewall system itself and of the routing, packet filtering, proxy, and logging capabilities – one of the primary test objectives.

Test both the implementation of the firewall system and the policy being implemented by the system.

It is not possible to exhaustively test all filtering rules and configurations. Build test cases and sample traffic that exercise each normal and excursion condition for each packet attribute such as protocol, source address, destination address, source port, and destination port. For example, a rule that permits TCP packets from any host to the public web server on port 80 should successfully permit or deny TCP packets to the web server at ports less than 80, port 80, and ports greater than 80.

<u>Design initial regression test suite</u>

Select a subset of test cases to be used for regression testing purposes during normal operations. These should include cases that verify that all incoming and outgoing packets are being routed, filtered, and logged as expected as well as service-specific cases that verify that packets requesting specific services (WWW, email, FTP, etc.) are being routed, filtered, and logged as expected.

Once the new firewall system is part of normal operations, you can use selected regression test cases to verify that a change does not affect operational capabilities that worked successfully prior to the change.

Acquire test tools

Candidate tool categories include network traffic generators, network monitors, portscanners, vulnerability detection tools, and intrusion detection systems.

Conduct Tests

Test in an isolated test environment

Exercising your installation and configuration procedures in a test environment will allow you to learn the requirements to efficiently install and configure both the operating system and your firewall software while minimizing the impact on your operational systems. It will highlight what, if any, hardware may be missing in your initial configuration.

Establish a test configuration so that your firewall system is interconnected between two isolated hosts, one playing the role of the external world and the other playing the role of your internal hosts. Ensure that the default gateway for the internal host is set to the firewall system under test. If you have implemented a central log host, place both the internal host and a log host on your internal network so that you can test logging options. If logging is performed on the firewall host, you can connect the internal host directly to the firewall host.

Have scanning or network sniffing tools in place on your outside and inside hosts to capture all traffic in both directions (inside to outside, outside to inside).

Perform the following steps:

- Disable packet filtering.
- Inject packets that will exercise all routing rules and send these through the firewall system.
- Ensure that packets are routed correctly by examining the firewall logs and your packet sniffer results.
- Turn on packet filtering.
- Inject network traffic that is an appropriate sampling of all possible source and destination IP addresses, across all ports, and for all protocols.
- Ensure that packets intended to be blocked (denied) are blocked. For example, if all UDP packets are to be blocked, ensure that none get through. Ensure that packets intended to enter or exit (permitted) do enter and exit. Do this by examining your firewall logs and scanner results.
- Ensure that packets intended for proxy handling are sent to the correct proxy and forwarded correctly.
- Scan for open and blocked ports to ensure your firewall system is performing as intended.
- Examine all of the network traffic that is logged and verify that the logging options associated with each packet filtering rule are operating as intended.
- Examine all of the network traffic that is logged and verify that the alert options associated with each logging option are sending alerts to the designated destination (such as the firewall administrator) using the specified mechanism (such as paging or email).

If possible, plan to conduct this step and the next step with at least two people: the original implementer of the routing configuration, packet filtering rules, proxies, logging options, and alert options, and an independent person who reviews what has been implemented, understands the intent, and agrees that the network topology and security policy have been reflected correctly.

Test in the production environment

Perform the following steps (this assumes you are migrating from a single layer to a multi layer firewall architecture):

- Connect your firewall system to your public and private networks.
- Set the routing configuration on selected public and private network hosts to direct traffic through the firewall system. The basis for selection is on a service-by-service basis, for example, the web server on your public network and the host storing the files that the web server needs to access on your private network. Cycle through the selection and exercise of all services such as web, file access, DNS, mail, and logging.
- Log the firewall system's incoming and outgoing network traffic. Use a scanner or network sniffer to observe what is happening.
- Ensure that packets intended to be blocked (denied) are blocked. Ensure that packets intended to enter or exit (permitted) do enter and exit.
- Scan all hosts in a selected portion of your network that includes the firewall system. Verify that you cannot gain any undesired information due to the scanning packets being blocked (denied). Attempt source port scanning using a well-known port such as the FTP-data port (port 20) to ensure that you cannot use the port for a service other than the one intended.
- You can use intrusion detection system tools in a simulated or live network traffic test to aid you in determining if your packet filtering rules and proxies are protecting your systems and networks from known attacks. You will need to run these tools for some period of time and review the results on a regular basis. You may want to defer this level of testing to normal operations once you have fully deployed the new firewall system.
- Examine all of the network traffic that is logged and verify that the logging options associated with each packet filtering rule and proxy are operating as intended.
- Examine all of the network traffic that is logged and verify that the alert options associated with each logging option are sending alerts to the designated destination (such as the firewall administrator) using the specified mechanism (such as paging or email).

You cannot do a final test of your routing configuration prior to connecting the firewall system to your operational external interfaces. As a result, you should run live packets through your internal networks using the new firewall system to the greatest extent possible prior to connecting to the outside world. To mitigate the risk of unexpected problems in this final test phase, consider initiating the operational connections for a small subset of hosts (such as those used by your system and firewall administrators) prior to connecting large numbers of user workstation or server hosts.

Test log file features

When log files are full, you need to select how the firewall system will respond. The possible options may be one or more of the following:

- shut down all external interfaces connected to the firewall
- continue to operate, overwriting the oldest entries
- continue to operate without logging

The first option is the preferred one but may not be available with all firewall products. Simulate a firewall log full condition and ensure that the firewall system behaves as expected based on the option you selected.

Select and exercise the appropriate settings for the archival of log files. The settings may include:

- log file destination (e.g., a local file on the firewall host or a central log file on a remote host)
- number of days before archiving a specified log file
- number of days before purging an archived version of a specified log file

Test failure modes

For each relevant failure mode described in the test plan, execute the test scenario causing or simulating the failure and exercise the mitigation strategy to see that it has the desired effect.

Scan for vulnerabilities

Use vulnerability detection tools to scan your firewall system to determine the presence of known vulnerabilities. If patches exist for vulnerabilities that a tool detects, install these on your firewall system and re-execute the tool to ensure that the vulnerability has been eliminated.

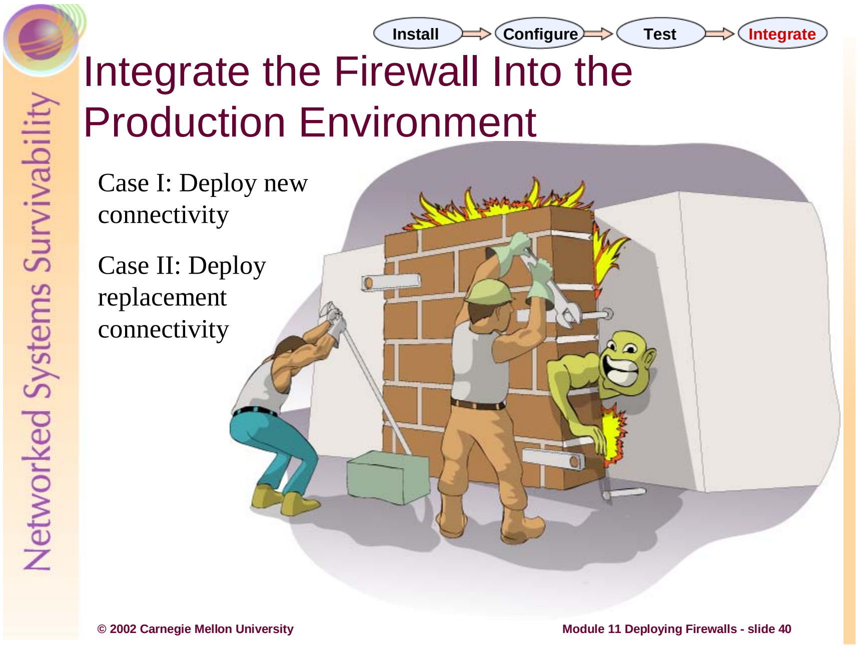
Prepare for Deployment and Operation

Prepare system for production use

Create and record cryptographic checksums or other integrity-checking baseline information of your firewall system and make a backup of your operational configuration once you have completed testing.

Prepare to perform ongoing monitoring

Given the complex nature of networks, their traffic, and firewall systems, ongoing monitoring is the only way to ensure that you have specified the correct security policy and that the policy is being implemented properly. Ensure that you have the necessary policies, procedures, tools, and staff resources in place to monitor your networks and systems including your firewall system [Allen 01].



Install → Configure → Test → Integrate

Networked Systems Survivability

Integrate the Firewall Into the Production Environment

Case I: Deploy new connectivity

Case II: Deploy replacement connectivity

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 40

There are a wide range of environments and configurations in which a firewall can be deployed. We consider two specific cases here to illustrate some of the integration issues that need to be addressed:

- The firewall system is being deployed to provide new physical connectivity between the networks that it was designed to interconnect. No previous firewall exists.
- The firewall system is being deployed as a replacement for an existing firewall system that already provides connectivity between these networks. This case also includes installing new logical connectivity on an existing firewall such as the addition of new ports or new services enabled by an updated policy

Case I: Deploy new connectivity

Shutdown the new firewall system, deploy it in the production environment, bring it back up carefully, and perform sample tests to prove it is forwarding and filtering as expected. Do not allow unfiltered traffic through the new firewall system during its physical installation into the production environment.

Make sure that you prepare the networks to be interconnected. Take into account, for example, IP addressing, routing, and DNS. Refer to [Cheswick 94] and [Zwicky 00] for additional information.

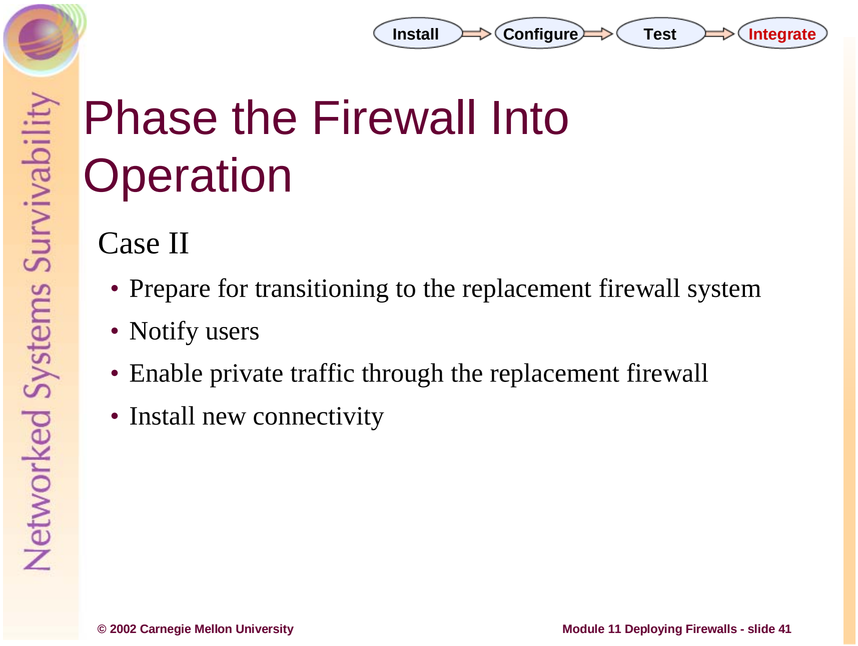
Consider making any new services available incrementally (a few at a time). The easiest way to accomplish this is to insert a “deny all” filter into each rule set immediately after the services you wish to make available. To make more services available, move the “deny all” rule further down in the rule sets until it gets to the bottom.

Case II: Deploy replacement connectivity

Deploy the replacement firewall system in parallel with the existing firewall system. Inserting the replacement firewall system into your production environment should not cause any changes to the environment, if possible.

Shutdown the replacement firewall system, deploy it in the production environment, bring it back up carefully, and perform sample tests to prove it is forwarding and filtering as expected. Do not allow unfiltered traffic through the replacement firewall system during its physical installation into the production environment.

During initial deployment, maintain the existing firewall system once it is disconnected. You can then switch back to it if the new system does not operate properly [Allen 01].



Networked Systems Survivability

Install → Configure → Test → Integrate

Phase the Firewall Into Operation

Case II

- Prepare for transitioning to the replacement firewall system
- Notify users
- Enable private traffic through the replacement firewall
- Install new connectivity

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 41

This description addresses Case II, phasing a replacement firewall into operation, where the new firewall system is replacing an existing firewall or router. It is the more challenging of the two cases and addresses issues that are also applicable to phasing a new firewall into operation.

Prepare for transitioning to the replacement firewall system

Once you have physically installed your tested, replacement firewall system into your production environment, you can then integrate it into your operational networks. In this configuration, no hosts are aware of the newly installed (replacement) firewall. Each host that is intended to send traffic through the firewall must be made aware of the replacement firewall's existence. You then make sure that the packet filters and proxies perform as expected in the production environment.

There are two sub-cases of Case II to consider

- the replacement firewall system is installed using the same IP addresses as the original system
- the replacement firewall system is installed using different IP addresses than the original system

If this operational scenario applies to your organization, refer to [Allen 01] for further details on alternatives for updating all hosts connected to the replacement firewall system with the routing information they each need to route traffic through this system.

You should maintain a fallback configuration to continue operations if the firewall system does not work as intended. Otherwise, you run the risk of incurring network outages that may affect the ability of your organization to conduct business that relies on internal and external communication via networks and through the firewall system. You should plan to perform the transition during non-business, non-peak hours such as over a weekend.

Notify users

Alert your users that the replacement firewall system is being brought into your operational environment. Inform them that the default gateways on their hosts will be changed to route network traffic through the firewall system and that this should be invisible to them. Indicate that they should inform their system or firewall administrator if they encounter any problems.

Enable private traffic through the replacement firewall

This case assumes that you are migrating from a single layer firewall architecture to a multiple layer architecture. The old firewall system becomes the interface to the external world and the new firewall system serves as the interface to your internal networks. This case also assumes that you have a network topology of one or more private networks and one or more public networks. The public networks typically connect hosts that respond to internal and external requests for service (such as WWW [HTTP], FTP, email [SMTP], and DNS). These hosts may also respond to internal requests for services such as SNMP and logging. The public network as described here can serve as your DMZ. The private network typically connects hosts that service your internal users including individual user workstations.

Configure and enable packets generated by the hosts on your private network to pass through the new firewall system:

1. Connect your public and private networks to the new firewall system.
2. Change the default gateway of the hosts on the private network from the old firewall system to the new firewall system.
3. Update the routing table on all public network hosts to route private network traffic through the new firewall system (vs. the old firewall system).
4. Disable the interface to the private network on the old firewall. Add a route to the old firewall system's routing table to route private network traffic through the new firewall system.
5. Ensure that traffic is being routed and filtered as expected.
6. Unplug the private network interface to the old firewall system.

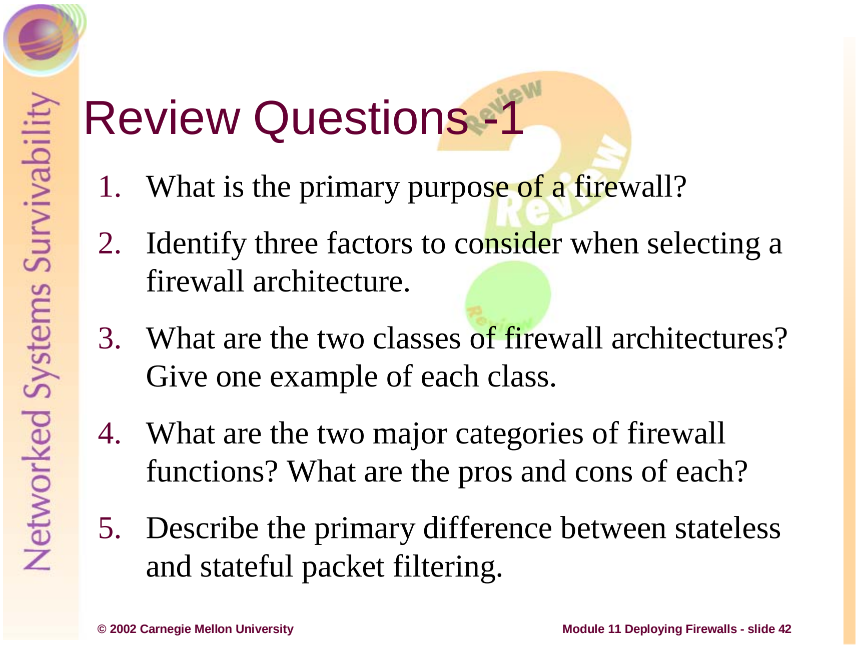
Refer to [Allen 01] Chapter 4 for additional details and example graphics.

Install new connectivity

If you are replacing an existing firewall system, maintain the physical connectivity through your existing system when you bring the new one online. This will allow you to determine if there are any hosts on your private network that have not had their routing configuration updated to interface with the new firewall system (as their traffic will continue to flow through the old firewall system). It will also allow you to make sure that everything is working as expected (i.e., all private network traffic is now being routed through the new firewall system).

If this does not work as intended:

1. Unplug the new firewall system hardware.
2. Plug the old firewall system hardware back into the private network.
3. Change the old firewall system’s private network interface address to the one that was used by the new firewall system, given all of your private network hosts are now using it.
4. Add the route for the private network in the old firewall system's routing table.



Networked Systems Survivability

Review Questions -1

1. What is the primary purpose of a firewall?
2. Identify three factors to consider when selecting a firewall architecture.
3. What are the two classes of firewall architectures? Give one example of each class.
4. What are the two major categories of firewall functions? What are the pros and cons of each?
5. Describe the primary difference between stateless and stateful packet filtering.

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 42

1. What is the primary purpose of a firewall?

   A firewall serves the purpose of implementing the access control policy governing the flow of traffic between two networks.

2. Identify three factors to consider when selecting a firewall architecture.

   - Understand and identify the boundaries between security domains
   - Identify the services you intend to offer and request from a public network
   - Anticipate system and network growth that the firewall will need to accommodate in the future

3. What are the two classes of firewall architectures? Give one example of each class.

   Single layer (basic with DMZ network) and multi layer (dual with DMZ network)

4. What are the two major categories of firewall functions? What are the pros and cons of each?

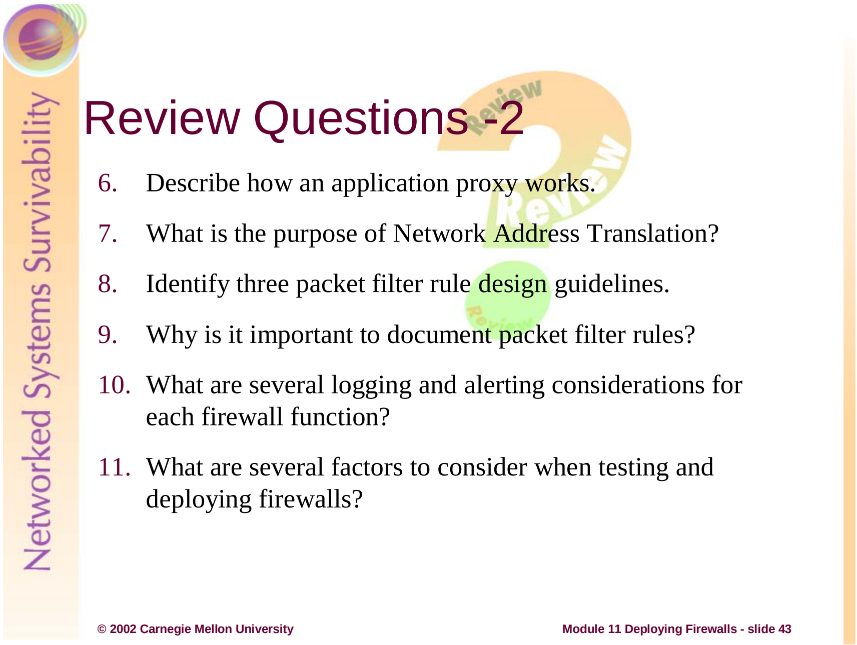
   Packet filters and proxies.

   Packet filters are fast, inexpensive, but can only filter on packet header information and rule sets can become complex.

   Proxies are typically slower, required for each type of service, but more secure, and can filter on packet contents as well as header information.

5. Describe the primary difference between stateless and stateful packet filtering.

   Stateless packet filtering processes each packet individually with no consideration for what has come before. Stateful packet filtering maintains a connection table and can permit or deny packets based on their legitimacy as part of an ongoing session. This aids in properly interpreting TCP status bits and handling connectionless protocols such as UDP and ICMP.



Networked Systems Survivability

Review Questions -2

6. Describe how an application proxy works.
7. What is the purpose of Network Address Translation?
8. Identify three packet filter rule design guidelines.
9. Why is it important to document packet filter rules?
10. What are several logging and alerting considerations for each firewall function?
11. What are several factors to consider when testing and deploying firewalls?

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 43

6. Describe how an application proxy works.

   An application proxy for a particular service (such as FTP, HTTP, mail, etc.) is a small software program that resides on a firewall system. It serves as the “go-between” that connects senders and receivers in that it receives all requests for a service and establishes the connection and communication with the service provider. This results in no direction communication between sender and receiver. An application proxy can screen, process, and manipulate packet header and content. A proxy must exist for each protocol or service where this type of filtering is required.

7. What is the purpose of Network Address Translation?

   NAT resides on a firewall system and translates between public IP addresses and internal network IP addresses, for both incoming and outgoing traffic. As a result, the only IP address that is externally visible is that of the firewall system. Using NAT provides the benefit of hiding internal network TCP/IP addresses and server/workstation configuration data and solves the problem of the need for externally-visible IP addresses for each internal host.

8. Identify three packet filter rule design guidelines.

   - Permit acceptable packets (per policy), deny everything else; explicitly include a “deny all” rule as the last rule
   - Use the IP address, not the host or domain name
   - Sort the rule set by protocol and then by port

9. Why is it important to document packet filter rules?

   Packet filter rules, as expressed in the syntax and semantics of a particular firewall vendor’s product, are often difficult to understand and can become complex. It is critical to capture how each rule traces to and enforces policy and to provide the rationale for the rule’s inclusion. In addition, rule sets are often combined to enhance performance and redundant rules are sometimes included to enhance logging. It is important to capture the purpose for such rules to avoid confusion and their inadvertent deletion.

10. What are several logging and alerting considerations for firewall systems and functions?

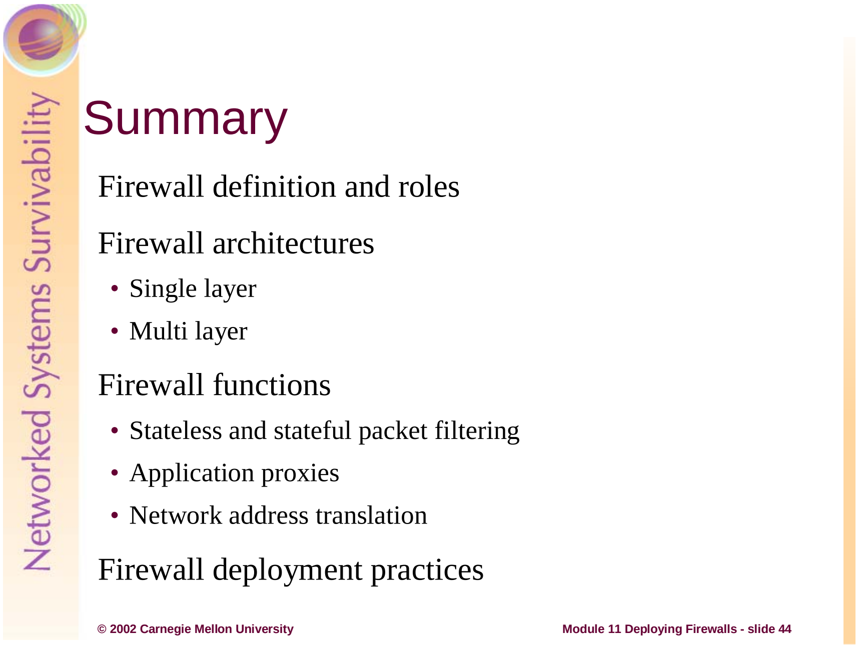
    Keep in mind that there is logging associated with packet processing and logging associated with firewall system operation. Typically, there are logging options and levels for each rule and for each type of function (packet filter, proxies).

    For alerts, determine what constitutes a significant event, the type of alerting mechanism to use (such as email displayed in a high priority window on the administrator's workstation, phone/voice mail, and/or pager notification), and the thresholds for generating an alert.

    Use a central log host that also houses a central alert mechanism if possible.

11. What are several factors to consider when testing and deploying firewalls?

    Make sure to create a test plan, test to the greatest extent possible in an isolated test environment before phasing the new firewall into production, exercise all log file features and firewall system failure modes, scan for vulnerabilities and patch as appropriate, design a regression test suite that can be used to ensure existing functions continue to work properly when changes are made, backup the firewall configuration when changes are made (and ensure you can recover from the backup), and generate cryptographic checksum information for the baseline configuration and each time the configuration changes.



Networked Systems Survivability

Summary

Firewall definition and roles

Firewall architectures

- Single layer
- Multi layer

Firewall functions

- Stateless and stateful packet filtering
- Application proxies
- Network address translation

Firewall deployment practices

© 2002 Carnegie Mellon University

Module 11 Deploying Firewalls - slide 44

References:

[Allen 01] Allen, Julia. *The CERT Guide to System and Network Security Practices*. Boston, MA: Addison-Wesley, 2001.

[Brenton 99] Brenton, Chris. *Mastering Network Security*. Alameda, CA: Sybex Network Press, 1999.

[Brenton 01] Brenton, Chris. “Firewalls 101: Perimeter Protection with Firewalls.” The Seventh Annual Conference on Securing Networks and Systems, San Diego, CA. SANS Institute, October 15-22, 2001.

[CBT 98] SmartForce Ireland Ltd. *Internet Security:Firewall Principles* Computer-Based Training course. CBT Systems Ltd., 1998.

[Cheswick 94] Cheswick, Willliam R., and Bellovin, Steven M. *Firewalls and Internet Security*. Reading, MA: Addison-Wesley, 1994.

[Comer 95] Comer, Douglas E. *Internetworking with TCP/IP, Vol. 1: Principles, Protocols, and Architecture*. 3rd edition. New York:” Prentice-Hall, 1995.

[Escamilla 98] Escamilla, Terry. *Intrusion Detection: Network Security Beyond the Firewall*. New York: Wiley Computer Publishing, 1998.

[Egevang 94] Egevang, K. Francis, P. *RFC 1631 The IP Network Address Translator (NAT)*. Internet Engineering Task Force Network Working Group, 1994. Online: http://www.ietf.org/rfc/rfc1631.txt

[ICSA 01] International Computer Security Association. *ICSA Labs Firewall Buyers Guide*. 2001. Available at http://www.icsalabs.com/html/communities/firewalls/buyers_guide2001/index.shtml. TruSecure’s/ICSA’s list of certified firewall products is available at http://www.trusecure.com/html/secsol/certifiedproducts.shtml#firewall.

[Lynch 00] Lynch, H. Merrill. “Firewall Fundamentals.” Information Systems Security. CRC Press, May/June 2000.

[NetIQ 01] NetIQ. “Reporting and Incident Management for Firewalls. November 8, 2001. Available at xxx.

[Ogletree 00] Ogletree, Terry William. *Practical Firewalls*. Que, June 2000.

[Ranum 98, Curtin 00] Ranum, Marcus J., Curtin, Matt. “Internet Firewalls: Frequently Asked Questions,” 1998, 2000. Available at http://www.interhack.net/pubs/fwfaq

[Smith 01] Smith, Gary. “A Brief Taxonomy of Firewalls – Great Walls of Fire.” May 18, 2001. Available at http://www.sans.org/infosecFAQ/firewall/taxonomy.htm

[Stevens 94] *TCP/IP Illustrated, Vol. 1: The Protocols*. Reading, MA: Addison-Wesley, 1994. Tyson, Jeff. “How Network Address Translation Works.” Online: http://www.howstuffworks/nat.htm and Cisco Systems Inc. “How NAT Works.” Online: http://www.cisco.com/warp/public/556/nat-cisco.shtml

[Zwicky 00] Zwicky, Elizabeth. Cooper, Simon. Chapman, D. Brent. *Building Internet Firewalls, 2d Edition*. Sebastopol, CA: O'Reilly & Associates, June 2000.