

# IS 2150 / TEL 2810

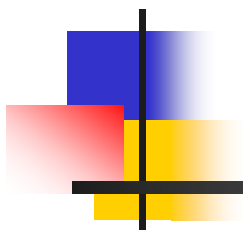
## Introduction to Security



James Joshi  
Professor, SIS

Lecture 12  
April 6, 2016

Intrusion Detection,  
Auditing System  
Firewalls & VPN



---

# Intrusion Detection



# Intrusion Detection/Response

---

- Denning:
  - Systems under attack fail to meet one or more of the following characteristics
    1. Actions of users/processes conform to statistically predictable patterns
    2. Actions of users/processes do not include sequences of commands to subvert security policy
    3. Actions of processes conform to specifications describing allowable actions



# Intrusion Detection

---

- Idea:
  - Attack can be discovered by one of the above being violated
- *Practical* goals of intrusion detection systems:
  - Detect a wide variety of intrusions (known + unknown)
  - Detect in a timely fashion
  - Present analysis in a useful manner
    - Need to monitor many components; proper interfaces needed
  - Be (sufficiently) accurate
    - Minimize *false positives* and *false negatives*



# IDS Types: Anomaly Detection

---

- Compare system characteristics with expected values
  - **Threshold metric:** statistics deviate / threshold
    - E.g., Number of failed logins
  - **Statistical moments:** mean/standard deviation
    - Number of user events in a system
    - Time periods of user activity
    - Resource usages profiles
  - **Markov model:** based on state, expected likelihood of transition to new states
    - If a low probability event occurs then it is considered suspicious



# IDS Types: Misuse Modeling

---

- Does sequence of instructions violate security policy?
  - Problem: How do we know all violating sequences?
- Solution: capture *known* violating sequences
  - Generate a rule set for an **intrusion signature**
- Alternate solution: State-transition approach
  - Known “bad” state transition from attack
  - Capture when transition has occurred (user → root)



# Specification Modeling

---

- Does sequence of instructions violate system specification?
  - What is the system specification?
- Need to formally specify operations of potentially critical code
  - *trusted* code
- Verify post-conditions met



# IDS Systems

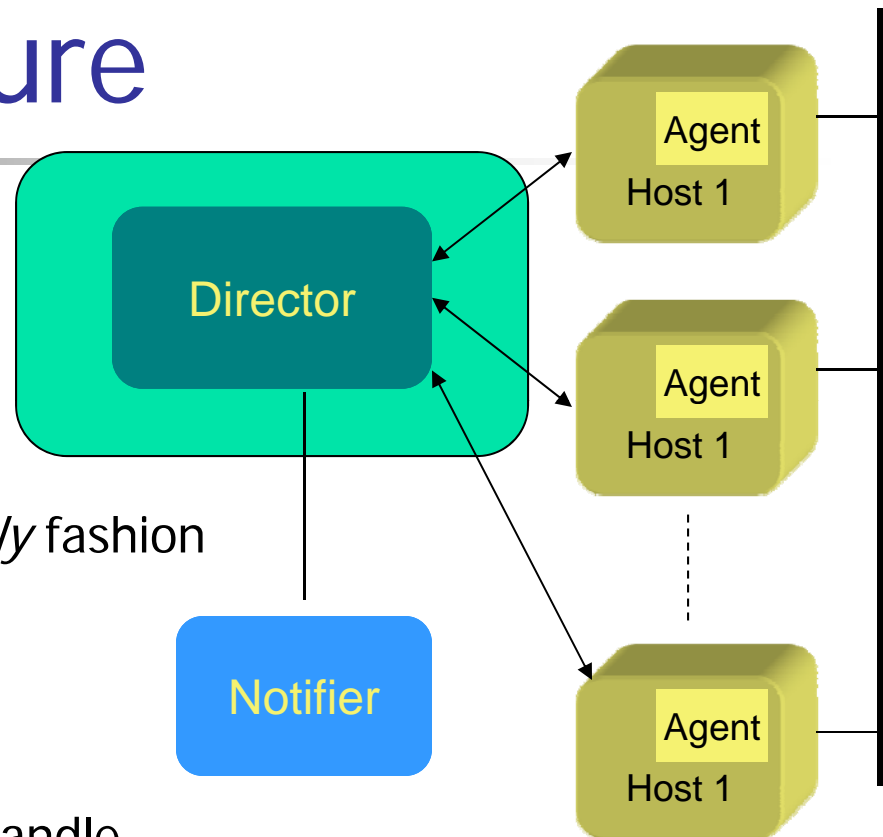
---

- Anomaly Detection
  - Intrusion Detection Expert System (IDES) – successor is NIDES
  - Network Security Monitor - NSM
- Misuse Detection
  - Intrusion Detection In Our Time- IDIOT (colored Petri-nets)
  - USTAT
  - ASAX (Rule-based)
- Hybrid
  - NADIR (Los Alamos)
  - Haystack (Air force, adaptive)
  - Hyperview (uses neural network)
  - Distributed IDS (Haystack + NSM)



# IDS Architecture

- Similar to Audit system
  - Log events
  - Analyze log
- Difference:
  - happens real-time - *timely* fashion
- (Distributed) IDS idea:
  - Agent generates log
  - Director analyzes logs
    - May be adaptive
  - Notifier decides how to handle result
    - GrIDS displays attacks in progress





# Where is the Agent?

---

- Host based IDS
  - watches events on the host
  - Often uses existing audit logs
- Network-based IDS
  - Packet sniffing
  - Firewall logs



# IDS Problem

---

- IDS useless unless accurate
  - Significant fraction of intrusions detected
  - Significant number of alarms correspond to intrusions
- Goal is
  - Reduce false positives
    - Reports an attack, but no attack underway
  - Reduce false negatives
    - An attack occurs but IDS fails to report



# Intrusion Response

---

- Incident Prevention
  - Stop attack before it succeeds
  - Measures to detect attacker
  - Example: Jailing (also Honeypots)
- Intrusion handling
  - Preparation for detecting attacks
  - Identification of an attack
  - Contain attack
  - Eradicate attack
  - Recover to secure state
  - Follow-up to the attack - Punish attacker



# Containment

---

- Passive monitoring
  - Track intruder actions
  - Eases recovery and punishment
- Constraining access
  - Downgrade attacker privileges
  - Protect sensitive information
  - **Why not just pull the plug**



# Eradication

---

- Terminate network connection
- Terminate processes
- Block future attacks
  - Close ports
  - Disallow specific IP addresses
  - Wrappers around attacked applications



# Follow-Up

---

- Legal action
  - Trace through network
- Cut off resources
  - Notify ISP of action
- Counterattack
  - Is this a good idea?



---

# Auditing





# What is Auditing?

---

- Auditing systems
  - Logging
  - Audit analysis
- Key issues
  - What to log?
  - What do you audit?
- Goals/uses
  - User accountability
  - Damage assessment
  - Determine causes of security violations
  - Describe security state for monitoring critical problems
  - Evaluate effectiveness of protection mechanisms



# Audit System Structure

---

- **Logger**
  - Records information, usually controlled by parameters
- **Analyzer**
  - Logs may come from multiple systems, or a single system
  - May lead to changes in logging
  - May lead to a report of an event
- **Notifier**
  - Informs analyst, other entities of results of analysis
  - May reconfigure logging and/or analysis on basis of results
  - May take some action



# Example: Windows NT

---

- Different logs for different types of events
  - *System event* logs record system crashes, component failures, and other system events
  - *Application event* logs record events that applications request be recorded
  - *Security event* log records security-critical events such as logging in and out, system file accesses, and other events
- Logs are binary; use *event viewer* to see them
- If log full, can have system shut down, logging disabled, or logs overwritten



# Windows NT Sample Entry

---

Date: 2/12/2000 Source: Security  
Time: 13:03 Category: Detailed Tracking  
Type: Success EventID: 592  
User: WINDSOR\Administrator  
Computer: WINDSOR

## Description:

A new process has been created:

New Process ID: 2216594592  
Image File Name:  
\Program Files\Internet Explorer\IEXPLORE.EXE  
Creator Process ID: 2217918496  
User Name: Administrator  
FDomain: WINDSOR  
Logon ID: (0x0,0x14B4c4)

[would be in graphical format]



# Designing an Audit System

---

- Goals determine what is logged
  - Idea: auditors want to detect violations of policy, which provides a set of constraints that the set of possible actions must satisfy
  - So, audit functions that may violate the constraints
- Constraint  $p_i : action \Rightarrow condition$



# Implementation Issues

---

- Show non-secure or find violations?
  - Former requires logging initial state and changes
- Defining violations
  - Does “write” include “append” and “create directory”?
- Multiple names for one object
  - Logging goes by *object* and not name
  - Representations can affect this
- Syntactic issues
  - Correct grammar – unambiguous semantics

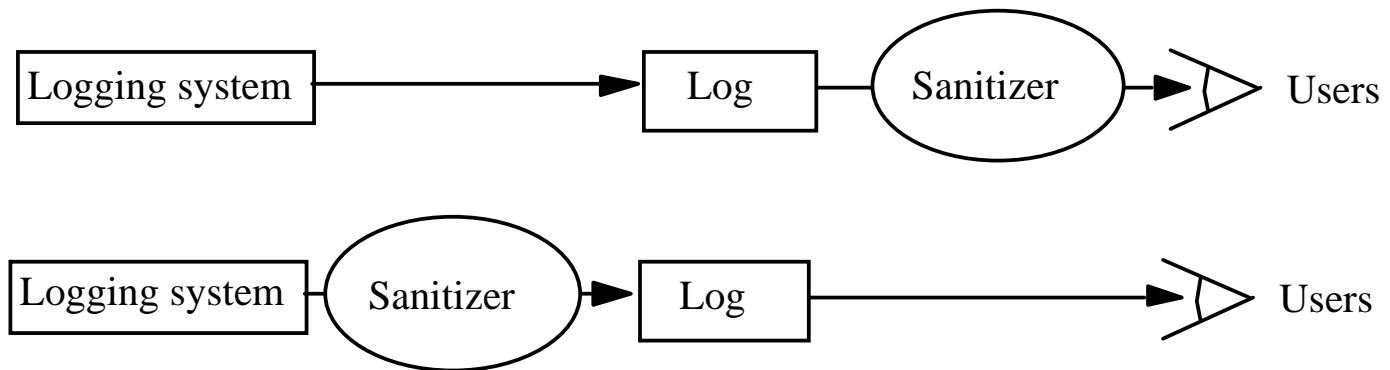


# Log Sanitization

---

- $U$  set of users,  $P$  policy defining set of information  $C(U)$  that  $U$  cannot see; log sanitized when all information in  $C(U)$  deleted from log
- Two types of  $P$ 
  - $C(U)$  can't leave site
    - People inside site are trusted and information not sensitive to them
  - $C(U)$  can't leave system
    - People inside site not trusted or (more commonly) information sensitive to them
    - Don't log this sensitive information

# Logging Organization



- Top prevents information from leaving site
  - Users' privacy not protected from system administrators, other administrative personnel
- Bottom prevents information from leaving system
  - Data simply not recorded, or data scrambled before recording (Cryptography)

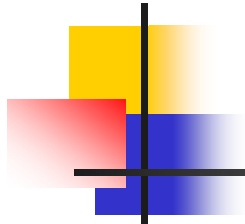




# Reconstruction

---

- *Anonymizing sanitizer* cannot be undone
- *Pseudonymizing sanitizer* can be undone
- Importance
  - Suppose security analysis requires access to information that was sanitized?



# Issue

---

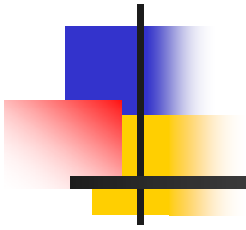
- Key: sanitization must preserve properties needed for security analysis
- If new properties added (because analysis changes), may have to resanitize information
  - This *requires* pseudonymous sanitization or the original log



# Example

---

- Company wants to keep its IP addresses secret, but wants a consultant to analyze logs for an address scanning attack
  - Connections to port 25 on IP addresses 10.163.5.10, 10.163.5.11, 10.163.5.12, 10.163.5.13, 10.163.5.14,
  - Sanitize with random IP addresses
    - Cannot see sweep through consecutive IP addresses
  - Sanitize with sequential IP addresses
    - Can see sweep through consecutive IP addresses



# Firewalls & VPN

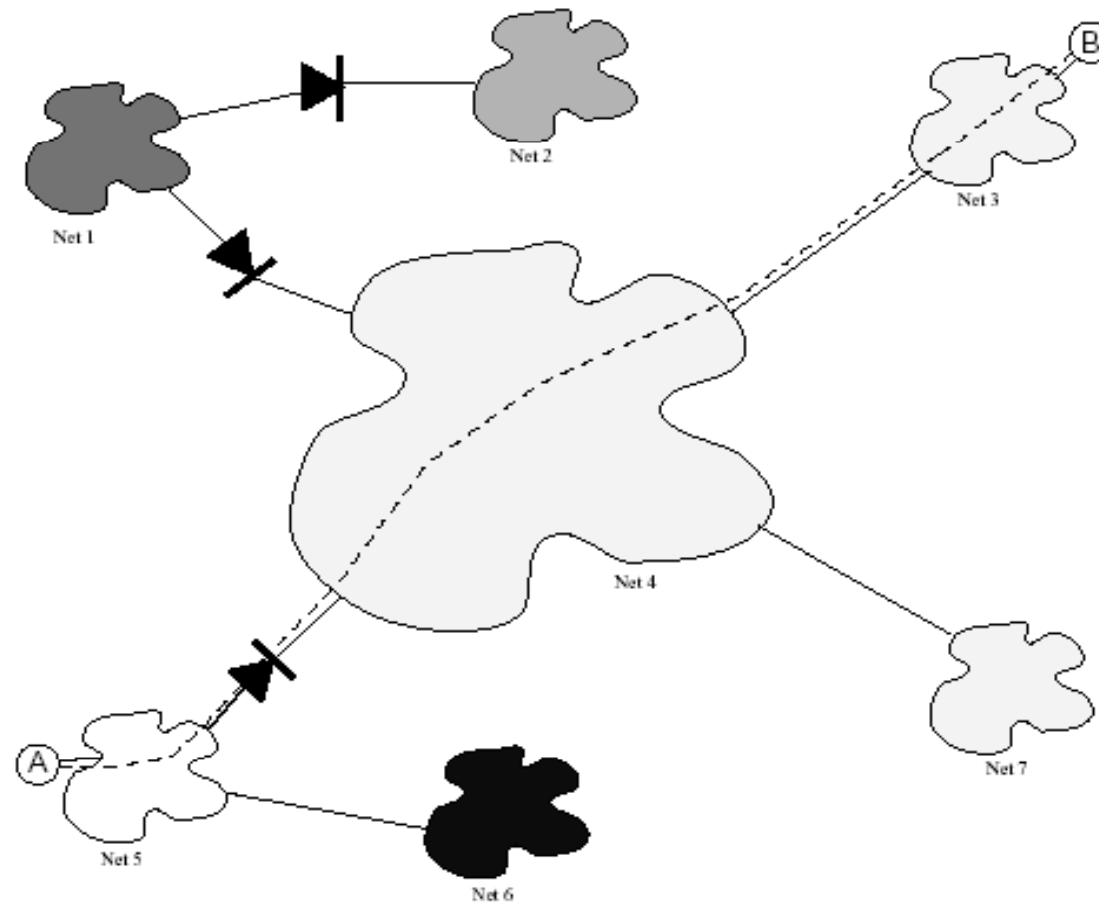


# What is a VPN?

---

- A network that supports a closed community of authorized users
  - Use the public Internet as part of the virtual private network
- There is traffic isolation
  - Contents, Services, Resources – secure
- Provide security!
  - Confidentiality and integrity of data
  - User authentication
  - Network access control
- IPSec can be used

# Tunneling in VPN





# Perimeter Defense

---

- Organizational system consists of a network of many host machines –
  - the system is as secure as the weakest link
- Use perimeter defense
  - Define a border and use gatekeeper (firewall)
- If host machines are scattered and need to use public network, use encryption
  - Virtual Private Networks (VPNs)



# Perimeter Defense

---

- Is it adequate?
  - Locating and securing all perimeter points is quite difficult
    - Less effective for large border
  - Inspecting/ensuring that remote connections are adequately protected is difficult
  - Insider attack is often the most damaging





# Firewalls

---

- Total isolation of networked systems is undesirable
  - Use firewalls to achieve selective border control
- Firewall
  - Is a configuration of machines and software
  - Limits network access
  - “for free” inside many devices
  - Alternate:  
a firewall is a host that mediates access to a network, allowing and disallowing certain type of access based on a configured security policy



# What Firewalls can't do

---

- They are not a panacea
  - Only adds to defense in depth
  - Can provide false sense of security
- Cannot prevent insider attack
- Firewalls act at a particular layer



# The Development of Firewalls

## First Generation

---

- Packet filtering firewalls
  - are simple networking devices that filter packets by examining every incoming and outgoing packet header
  - can selectively filter packets based on values in the packet header
    - IP address, type of packet, port request, and/or other elements



# Second Generation

---

- Application-level firewalls
  - often consists of dedicated computers kept separate from the first filtering router (edge router)
    - Commonly used in conjunction with a second or internal filtering router
  - Proxy server, rather than the Web server, is exposed to outside world from within a network segment called the demilitarized zone (DMZ),
- Implemented for specific protocols



# Third Generation

---

- Stateful inspection firewalls
  - keep track of each network connection established between internal and external systems
    - state and context of each packet exchanged (who / when)
  - can restrict incoming packets by matching with requests from internal hosts
  - Non-matching packets - it uses ACL rights to determine whether to allow the packet to pass



# Fourth Generation

---

- A fourth-generation firewall, or dynamic packet filtering firewall,
  - allows only a particular packet with a specific source, destination, and port address to pass through the firewall
  - understands how the protocol functions, and by opening and closing pathways in the firewall
  - an intermediate form,
    - between traditional static packet filters and application proxies



# Firewall Architectures

---

- For each type –
  - can be implemented in a number of architectural configurations
- Four architectural implementations of firewalls are especially common:
  - Packet filtering routers
  - Screened-host firewalls
  - Dual-homed host firewalls
  - Screened-subnet firewalls



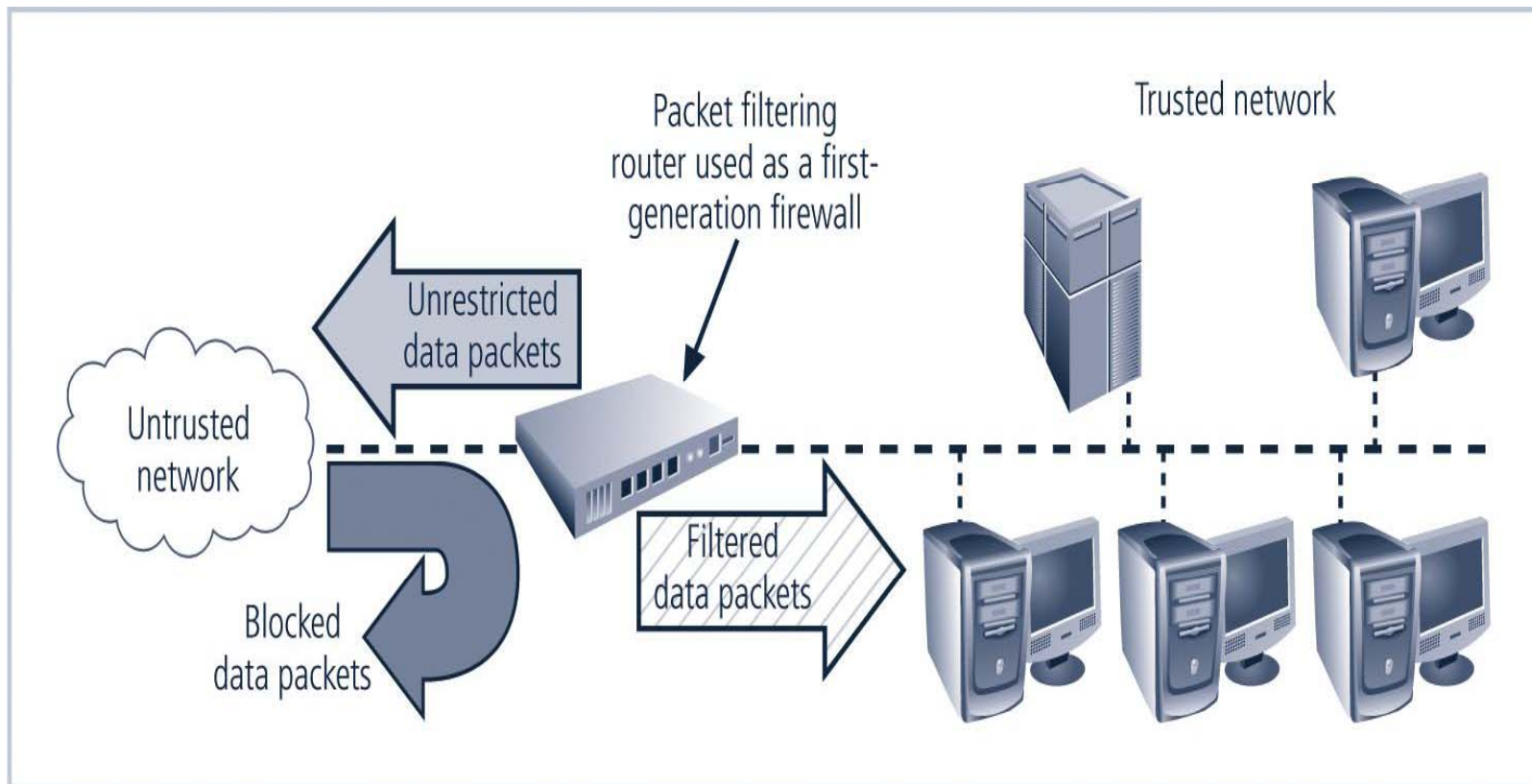
# Packet Filtering Routers

---

- Most organizations with an Internet connection
  - use a router between their internal networks and the external service provider
- Limitation
  - lacks auditing and strong authentication
  - complexity of the ACLs used to filter the packets can grow to the point of degrading network performance



# Packet Filtering Router/Firewall



**FIGURE 9-5** Packet Filtering Firewall

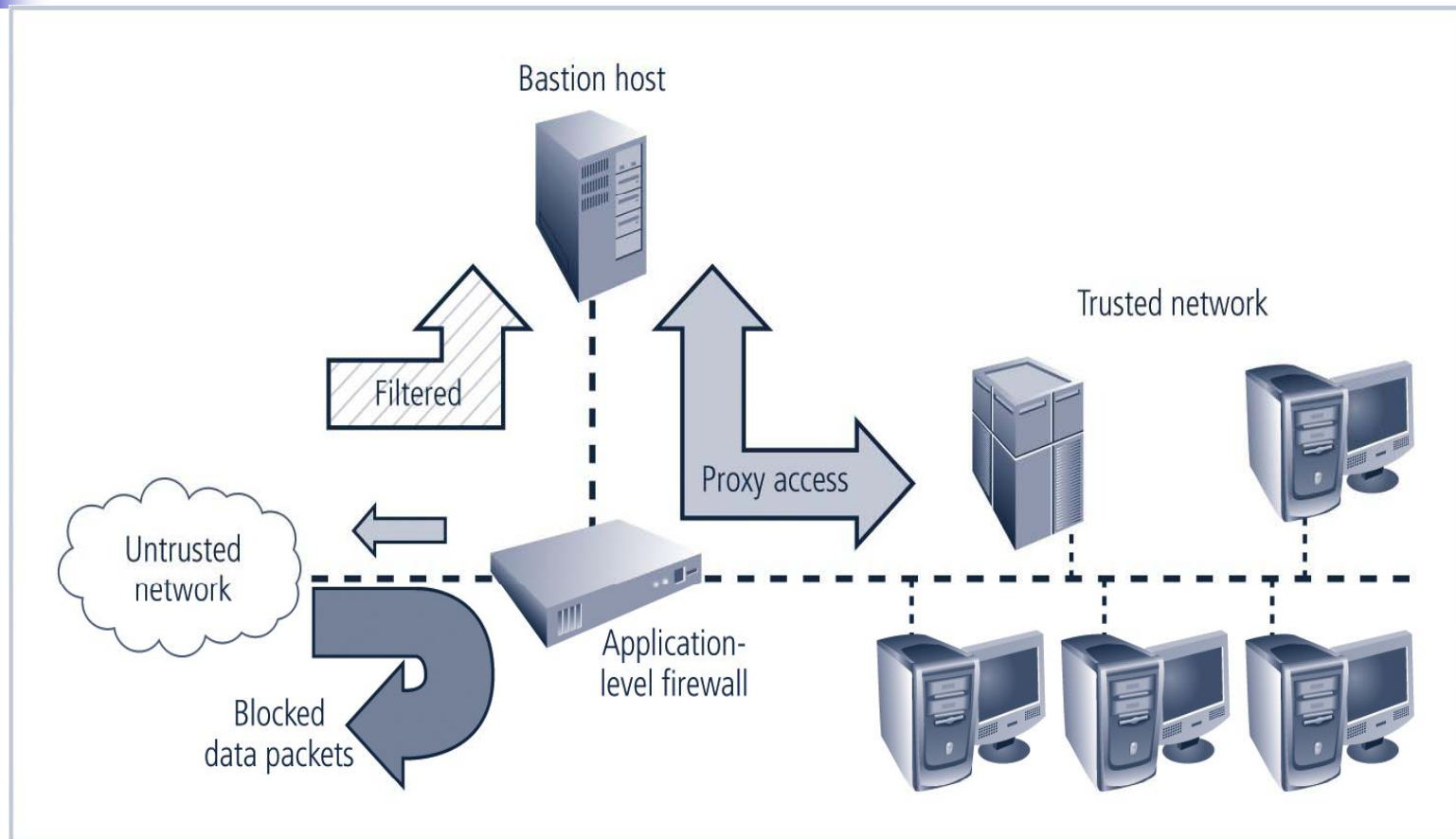


# Screened-Host Firewall Systems

---

- Screened-host firewall systems
  - combine packet filtering router with a separate, dedicated firewall such as an application proxy server
  - Application proxy examines an application layer protocol, such as HTTP, and performs the proxy services
  - This separate host, referred to as a **bastion host**, represents a single, rich target for external attacks, and should be very thoroughly secured

# Screened-Host Firewall



**FIGURE 9-6** Screened-Host Firewall



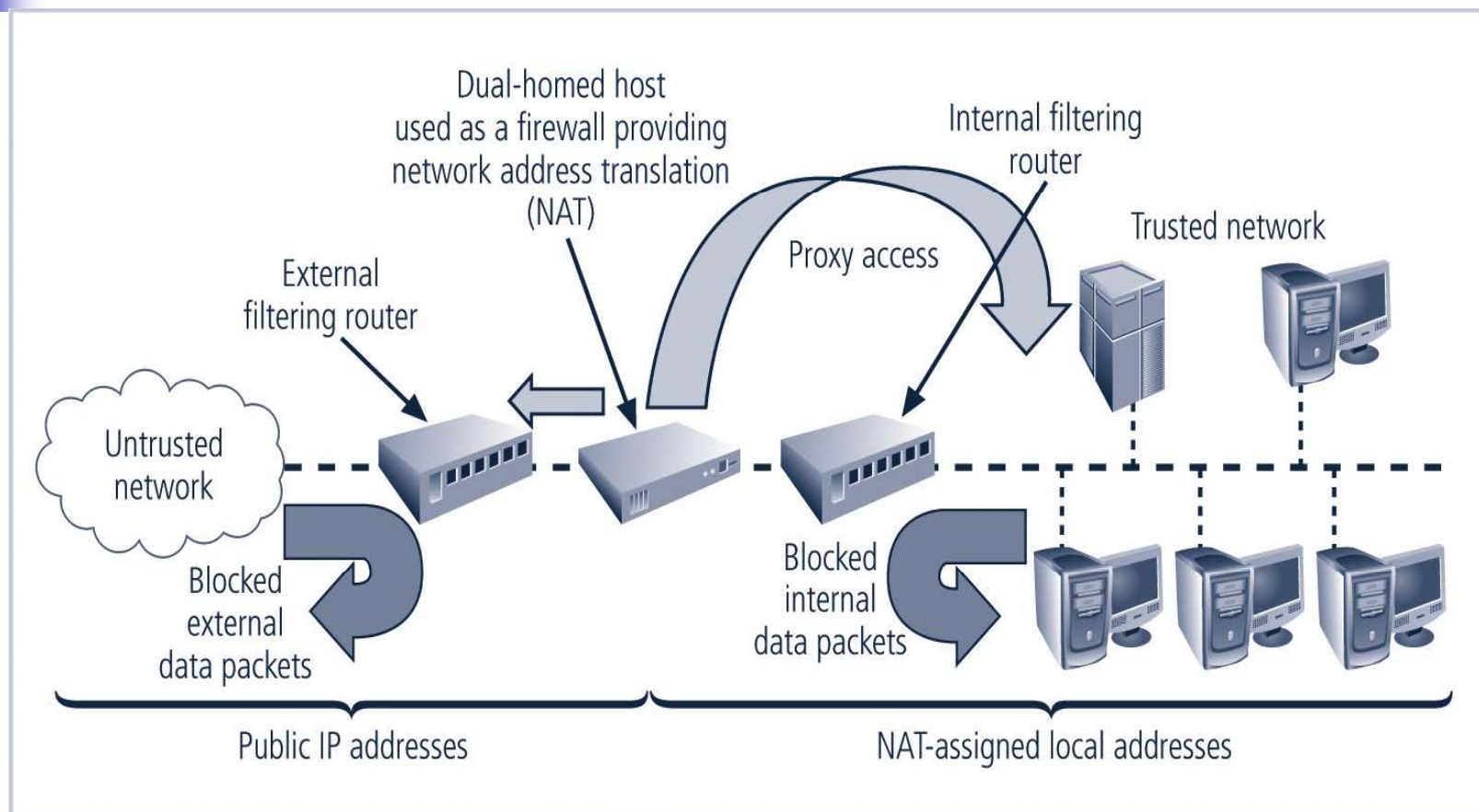
# Dual-Homed Host Firewalls

---

- In this configuration, the bastion host contains two network interfaces:
  - One connected to external network
  - One connected to internal network, requiring all traffic to travel through the firewall to move between the internal and external networks
- Network–address translation (NAT) is often implemented with this architecture
  - Converts external IP addresses to special ranges of internal IP addresses

# Figure 9-7

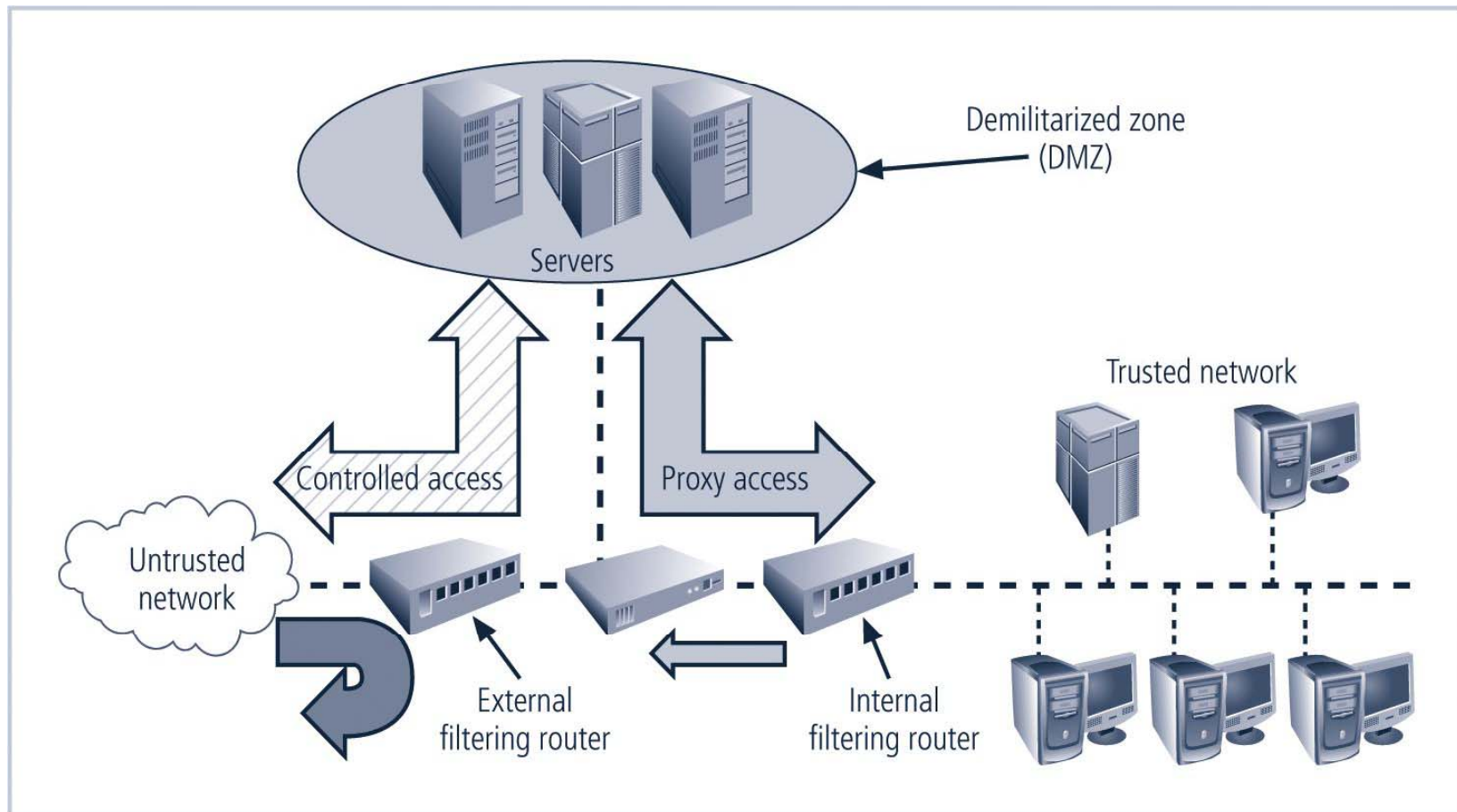
## Dual-Homed Host Firewall



**FIGURE 9-7** Dual-Homed Host Firewall

# Screened-Subnet Firewalls (with DMZ)

consists of one or more internal bastion hosts located behind a packet filtering router, with each host protecting the trusted network





# Summary

---

- Intrusion detection system
  - Various types
- Auditing
  - Design issues
- Firewalls
  - Four different types