

## Firewall

Firewall

1

## Why Firewalls?

- **Internet connectivity** is no longer an option for most corporations
- The Internet allows you access to worldwide resources, but...  
...the Internet also allows the *world* to try and access your resources
- This is a **grave risk** to most organizations

Firewall

2

## Cosa sono i Firewalls?

- Un **firewall** è un sw inserito tra la rete proprietaria e Internet
- Stabilisce un **perimetro**
- Fornisce un punto fisso in cui si possono imporre proprie politiche di sicurezza
- Le funzioni di firewall possono essere realizzate da un singolo Sistema o da più sistemi

Firewall

3

## Good Fences Make Good Neighbors – Robert Frost, “Mending Wall”



Firewall

4

## Design Goals

- **All traffic**, from inside to outside and vice versa, must pass **through the firewall**
- Only **authorized traffic** (defined by the **security policy**) is allowed to flow
- Firewall is **immune to penetration** – uses a trusted system

## Access Control Techniques

- **Service Control** – types of Internet service accessed inbound and outbound
- **Direction Control** – direction in which particular services may be initiated
- **User Control** – access to a service is controlled according to users
- **Behavior Control** – controls how particular services are used

## Scope of Firewalls

- **Single choke point** - to protect vulnerable services from various kinds of attack (spoofing, DOS)
- **Singular monitoring point** – location for monitoring, auditing and event triggering

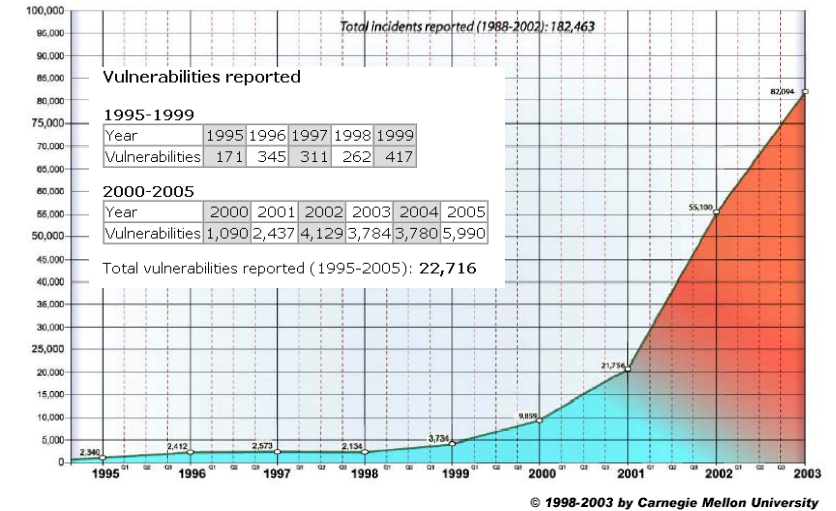
## Scope of Firewalls

- **Platform for non-security functions** – can be used for network address translation and network management
- **Platform for IPSec** – implements VPN via tunnel mode

## Limitazioni dei Firewall

- NON protegge da attacchi capaci di attraversare il firewall stesso – **bypass attack**
- NON protegge da **internal threats**
- NON protegge da programme infetti da **virus**

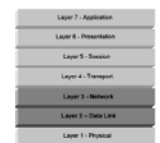
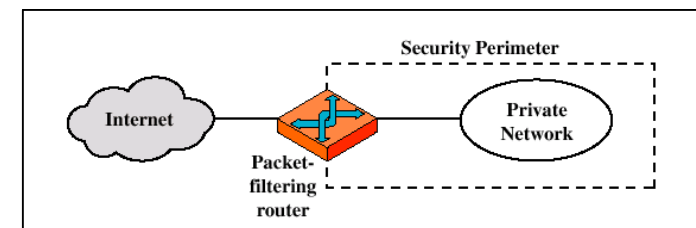
## CERT/CC Incidents Reported



## Types of Firewalls

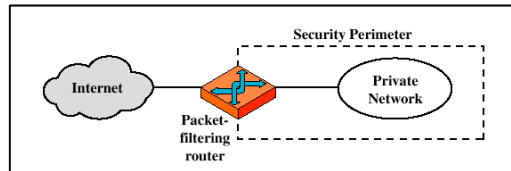
- **Packet Filtering Router**
- **Application Level Gateway**
- **Circuit Level Gateway**

## Packet Filtering



## Packet Filtering Router

- Applica un insieme di regole ai pck in ingresso e li lascia passare in funzione del risultato
- Filtra i pck in entrambe le direzioni

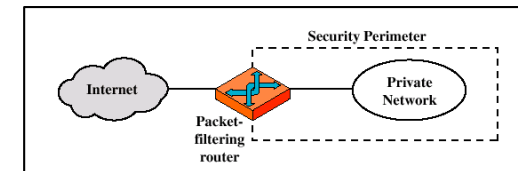


Firewall

13

## Packet Filtering Router

- Le regole sono basate sugli indirizzi di *source* e *destination* e sui *port* number
- *List of rules* looking for a match
- If no match, *default* action is taken



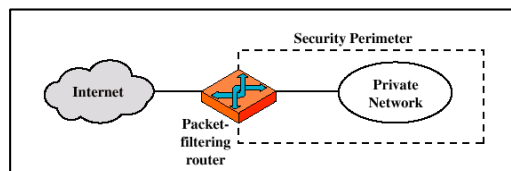
Firewall

14

## Packet Filtering Router

Two default policies:

- **default = discard:**  
*That which is not expressly permitted is prohibited*
- **default = forward:**  
*That which is not expressly prohibited is permitted*



Firewall

15

## Packet Filtering

- *Vantaggio:* concettualmente semplice, trasparente e molto veloce
- *Svantaggio:* difficoltà nella corretta definizione delle regole setting

Firewall

16

# Packet Filtering Attacks

- **IP address spoofing** – packets from the outside have internal addresses in their source IP address field
- **Source routing attacks** – route of packet is specified to bypass security measures
- **Tiny fragment attack** – designed to circumvent filtering rules that depend on TCP header information

# Real Life Example

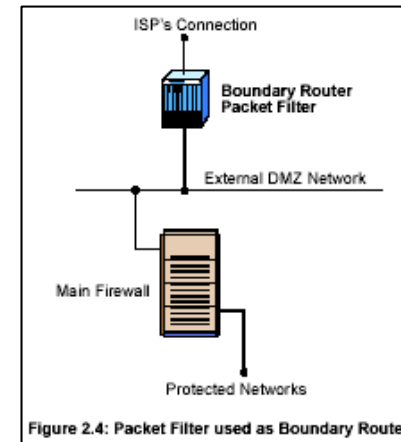
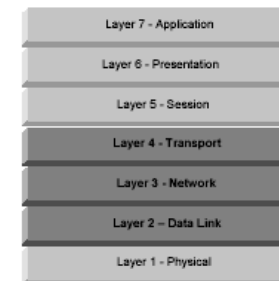


Figure 2.4: Packet Filter used as Boundary Router

# Real Life Example

|   | Source Address | Source Port | Destination Address | Destination Port | Action | Description   |
|---|----------------|-------------|---------------------|------------------|--------|---|
| 1 | Any            | Any         | 192.168.1.0         | > 1023           | Allow  | Rule to allow return TCP Connections to internal subnet                   |
| 2 | 192.168.1.1    | Any         | Any                 | Any              | Deny   | Prevent Firewall system itself from directly connecting to anything       |
| 3 | Any            | Any         | 192.168.1.1         | Any              | Deny   | Prevent External users from directly accessing the Firewall system.       |
| 4 | 192.168.1.0    | Any         | Any                 | Any              | Allow  | Internal Users can access External servers                                |
| 5 | Any            | Any         | 192.168.1.2         | SMTP             | Allow  | Allow External Users to send email in                                     |
| 6 | Any            | Any         | 192.168.1.3         | HTTP             | Allow  | Allow External Users to access WWW server                                 |
| 7 | Any            | Any         | Any                 | Any              | Deny   | "Catch-All" Rule - Everything not previously allowed is explicitly denied |

# Stateful Inspection



Layers Addressed By Stateful Inspection

## Stateful Inspection

- Inbound connections are **above port 1023**
- Solve this problem by creating a **directory of outbound TCP connections**, along with each session's corresponding high-numbered client port
- **State Table** - used to validate any inbound traffic

Firewall

21

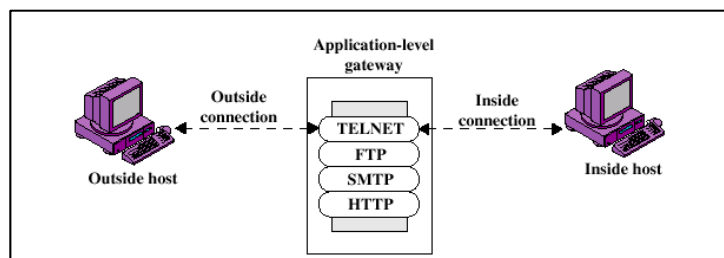
## Stateful Inspection

- **More secure** because the firewall tracks client ports individually rather than opening all high-numbered ports for external access.
- Adds **Layer 4 awareness** to the standard packet filter architecture.
- Useful or applicable **only** within **TCP/IP** network infrastructures
- **Superset of packet filter** firewall functionality

Firewall

22

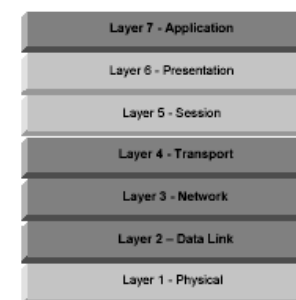
## Application Level Gateway



Firewall

23

## Application Gateway Firewalls



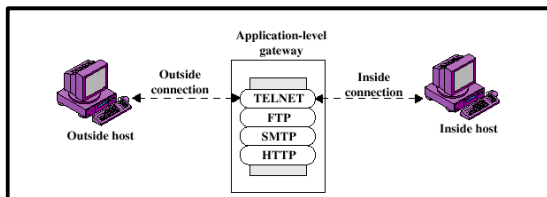
Layers Addressed by  
Application-Proxy Gateway Firewalls

Firewall

24

# Application Level Gateway

- Acts as a **relay** of application level traffic
- Also called a **proxy**
- User contacts gateway for TELNET to remote host, user is authenticated, then gateway contacts remote host and relays info between two end points

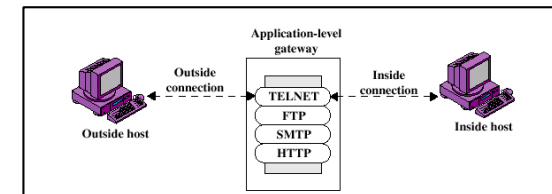


Firewall

25

# Application Level Gateway

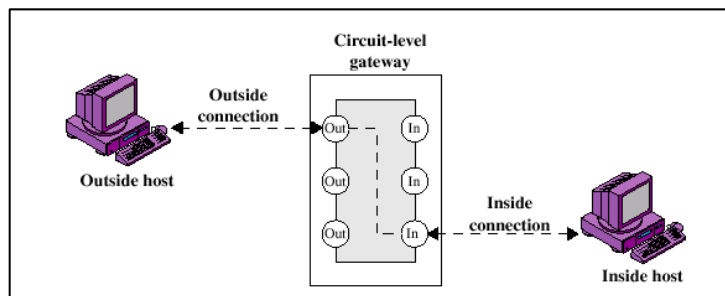
- If proxy code for application is not supported, **no forwarding** of packets
- Can **examine the packets** to ensure the security of the application – **full packet awareness**
- Very **easy to log** since entire packet seen
- Disadvantage:** additional processing overhead for each connection – increase load



Firewall

26

# Circuit-Level Gateway

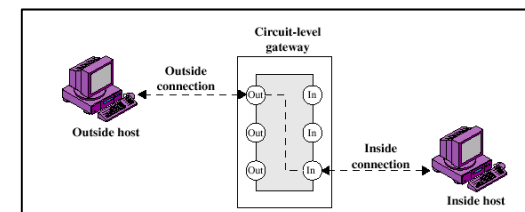


Firewall

27

# Circuit Level Gateway

- Does not** permit an end-to-end TCP connection
- Sets up **two TCP connections** one between itself and a TCP user on the inside and one between itself and a TCP user on the outside
- Relays TCP segments** from one connection to the other **without examining the contents**

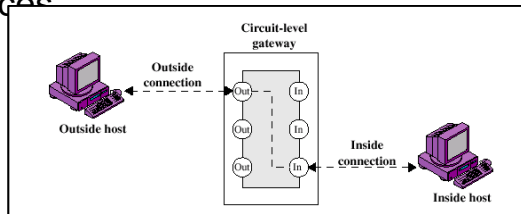


Firewall

28

## Circuit Level Gateway

- *Security function* (implements policy) determines *which connections will be allowed*
- Used where *internal users are trusted* for all outbound services
- Often *combined with a proxy* for inbound services

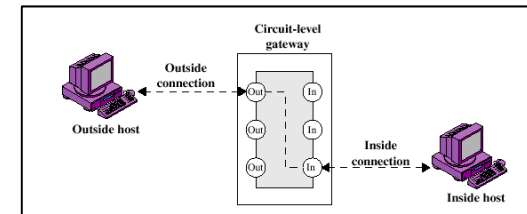


Firewall

29

## Circuit Level Gateway

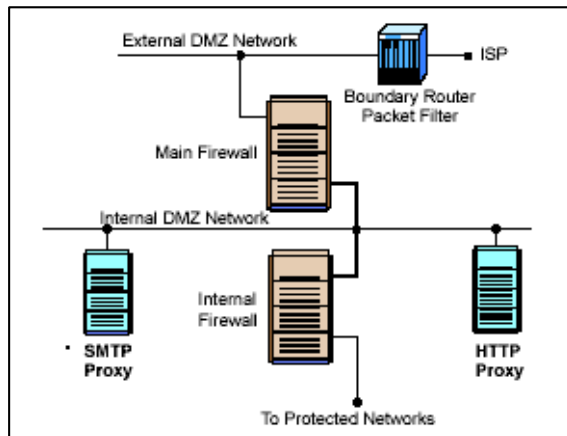
- **SOCKS** package V5 – RFC 1928
- **Shim** between application and transport layers
- Uses port 1080
- Requires *SOCKS-ified client*
- *Disadvantage*: some implementations require a special client



Firewall

30

## Dedicated Proxy Servers



Firewall

31

## Hybrid Firewalls

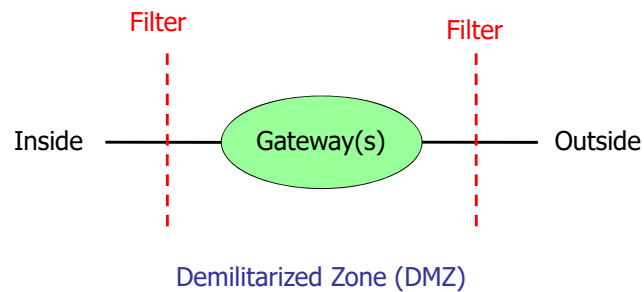
- “*blurring of lines*” that differentiate types of firewalls
- Application proxy gateway firewall vendors have implemented basic packet filter functionality in order to provide better support for UDP based applications
- Stateful inspection packet filter firewall vendors have implemented basic application proxy functionality to offset some of the weaknesses associated with packet filtering

Firewall

32

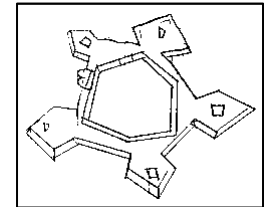


## Schematic of a Firewall



## Bastion Host

- *Exposed* gateway is called the **bastion host**
- Sits in the *DMZ*
- Usually a platform for an application or circuit level gateway
- Hardened, *trusted system*
- Only essential services



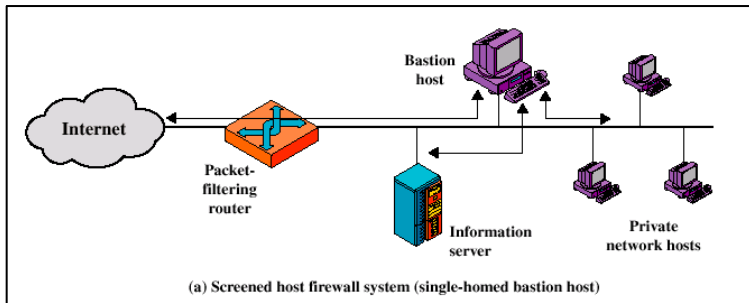
## Bastion Host

- Allows *access only* to *specific hosts*
- Maintains detailed *audit information* by logging all traffic
- *Choke point* for discovering and terminating intruder attacks
- Each proxy is a *small, highly secure network software package* that is a subset of the general application

## Bastion Host

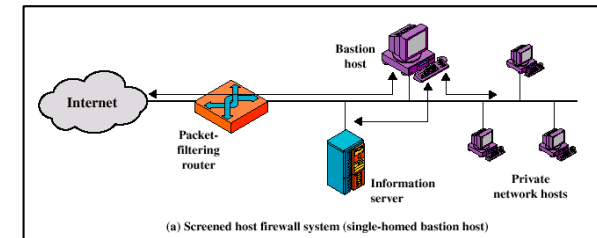
- *Proxies* on bastion host are *independent* of each other
- *No disk access* other than to read initial configuration
- Proxies *run* as *non-privileged* users
- *Limited access* to bastion host

# Bastion Host, Single-Homed



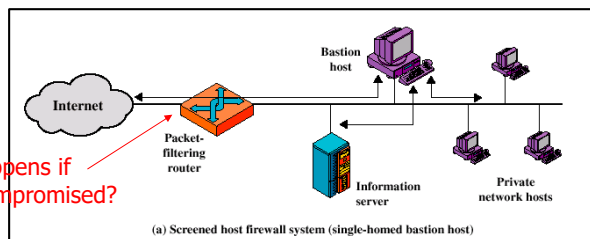
# Bastion Host, Single-Homed

- *Two systems*: packet filtering router and bastion host
- For traffic from the *Internet*, only IP packets *destined* for the *bastion* host are allowed
- For traffic from the *internal network*, only relayed packets *from* the *bastion* host are allowed out



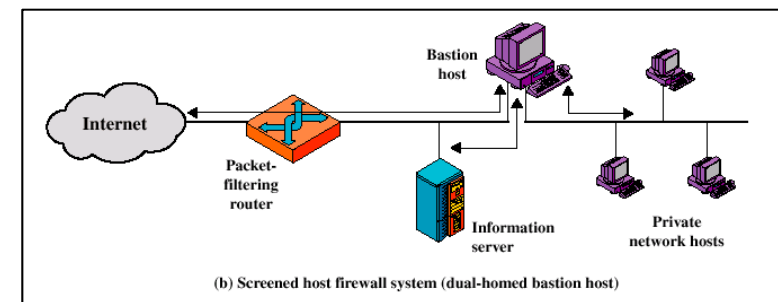
# Bastion Host, Single-Homed

- Bastion host *performs authentication*  
Implements *both* packet level and application level filtering
- Intruder *penetrates two separate systems* before internal network is compromised
- May contain a *public information* server



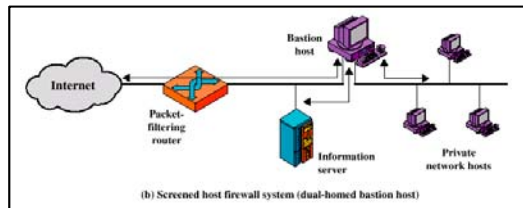
What happens if this is compromised?

# Bastion Host, Dual-Homed



## Bastion Host, Dual-homed

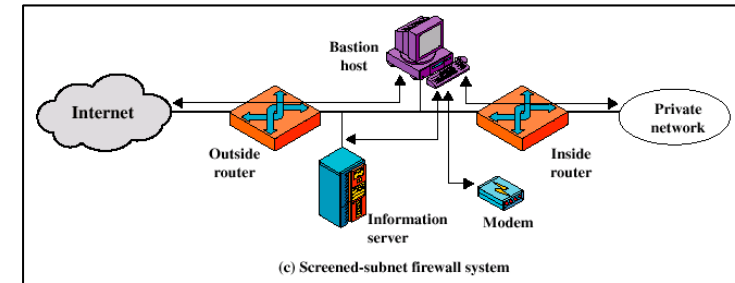
- Bastion host *second defense layer*
- **Internal network** is completely **isolated**
- **Packet forwarding** is turned **off**
- More secure



Firewall

41

## Screened Subnet

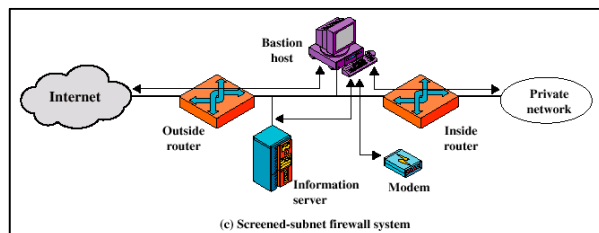


Firewall

42

## Screened Subnet

- Most secure
- **Isolated subnet** with bastion host between two packet filtering routers
- **Traffic** across screened subnet is **blocked**
- Three **layers** of defense
- Internal network is **invisible** to the Internet



Firewall

43

## DMZ Building Guidelines

- **Keep It Simple** - KISS principle - the more simple the firewall solution, the more secure and more manageable
- **Use Devices as They Were Intended to Be Used** – don't make switches into firewalls
- **Create Defense in Depth** – use layers, routers and servers for defense
- **Pay Attention to Internal Threats** – “crown jewels” go behind internal firewall – adage: “all rules are meant to be broken”

Firewall

44

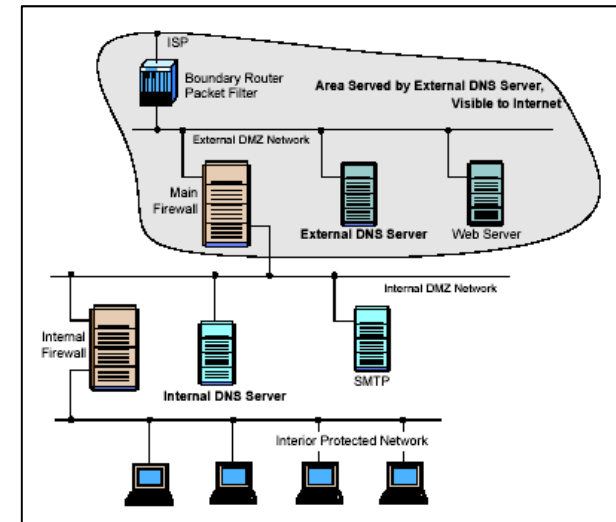
## Taming the DNS

- Need *two* DNS servers
- Don't want to reveal internal names and addresses
- Internal network has an isolated, pseudo-root DNS
- *Forwards* requests to the external DNS
- "Split DNS" or "Split Brain"

Firewall

45

## Taming the DNS



Firewall

46

## Network Address Translation

- Solves address depletion problems with IPv4
- RFC 2663 – IP Network Address Translator Terminology and Considerations, 1996
- Gateways to disparate networks
- Hides internal addresses
- Port Address Translation (PAT) – a variation using ports

Firewall

47

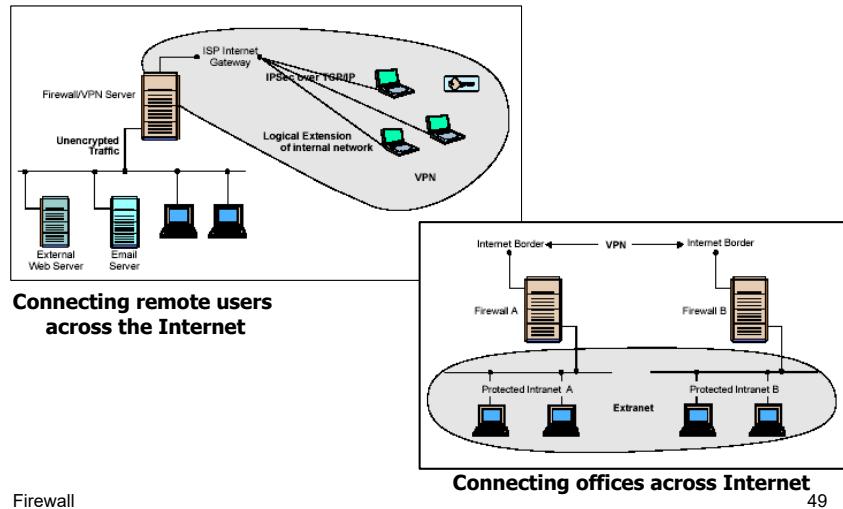
## Secure Shell (SSH)

- Eliminates "Crunchy Cookie" DMZ
- Everything is encrypted
- Used for system administration and remote access
- SSH2 – [www.ssh.com](http://www.ssh.com)

Firewall

48

## VPN's Another Type of Firewall



Firewall

49

## Other Types Of Firewalls

- **Host Based Firewalls** – comes with some operating systems (LINUX, WIN/XP) – **ipfilter** is a popular one

<http://coombs.anu.edu.au/~avalon/>

Firewall

50

## Other Types Of Firewalls

- **Personal Firewalls Appliances**  
– personal firewall appliances are designed to protect small networks such as networks that might be found in home offices
- **Provide:** print server, shared broadband use, firewall, DHCP server and NAT



(NB: This is not an endorsement of any product)

Firewall

51

## Network Security

### Trusted Systems

Firewall

52

## Access Matrix

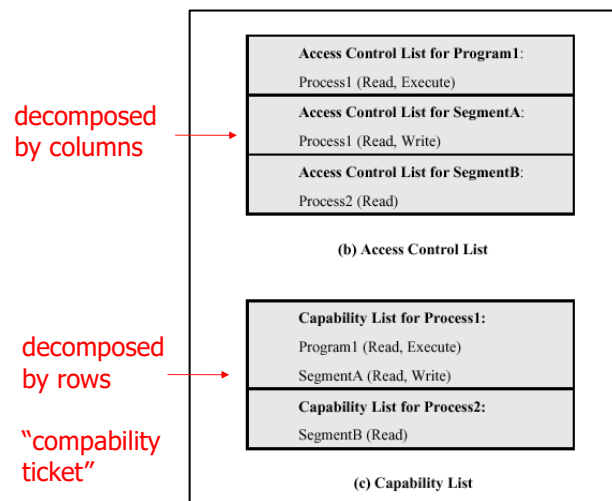
General model of access control:

- **Subject** – entity capable of accessing objects (user = process= subject)
- **Object** – anything to which access is controlled (files, programs, memory)
- **Access right** – way in which an object is accessed by a subject (read, write, exe)

## Access Matrix

|          | Program1        | ... | SegmentA      | SegmentB |
|----------|-----------------|-----|---------------|----------|
| Process1 | Read<br>Execute |     | Read<br>Write |          |
| Process2 |                 |     |               | Read     |
| •        |                 |     |               |          |
| •        |                 |     |               |          |
| •        |                 |     |               |          |

## Access Control List



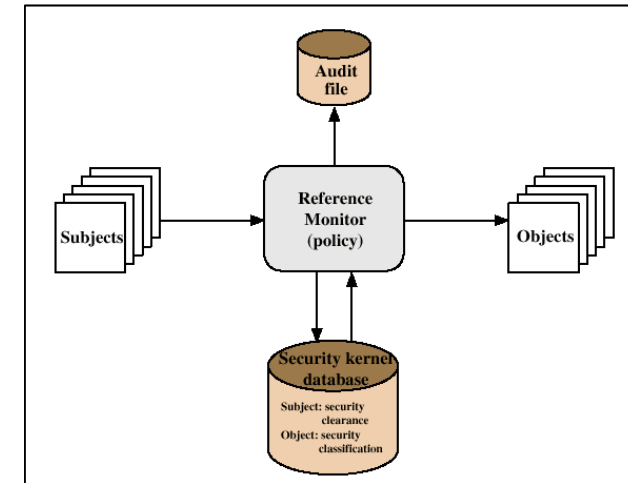
## Concept of Trusted Systems

- We've been **concerned** with **protecting** a message from active or passive attack **by given user**
- Different requirement is to **protect** data or resources on the basis of **security levels** (unclassified, confidential, secret and top secret)

## Concept of Trusted Systems

- **Multilevel security** – subject at a high level may not convey information to a subject at a lower or non-comparable level unless that flow accurately reflects the will of an authorized user
- **No read up:** Subject can only read an object of less or equal security level
- **No write down:** Subject can only write into an object of greater or equal security level

## Reference Monitor



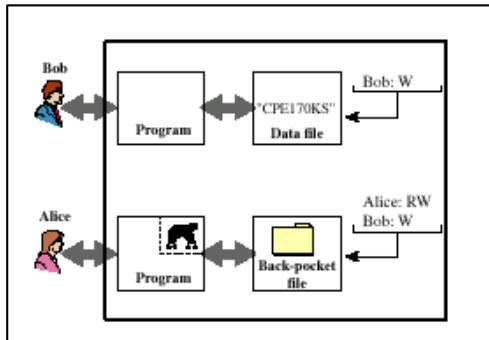
## Reference Monitor

- **Reference monitor** is a controlling element in hardware and OS
- Enforces the security rules in the **security kernel database** (no read up, no write down)

## Trusted System Properties

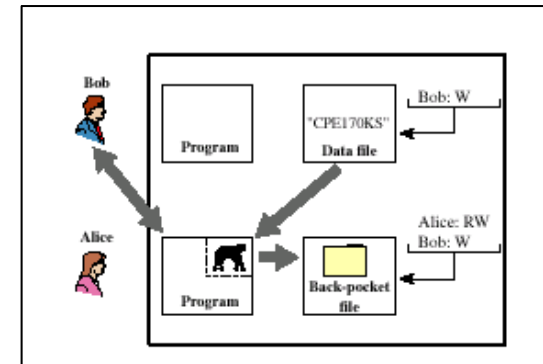
- **Complete mediation** – security rules enforced on every access
- **Isolation** – reference monitor and database are protected from unauthorized modification
- **Verifiability** – reference monitor's correctness must be mathematically provable

## Trojan Horse Defense



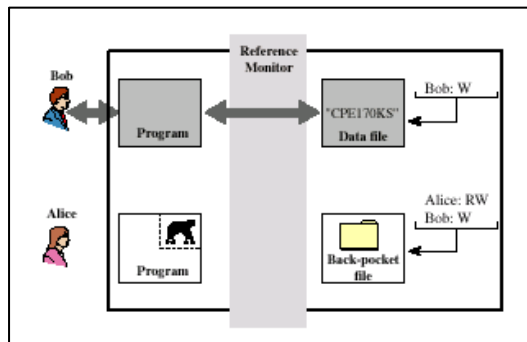
Alice installs trojan horse program and gives Bob write only permission

## Trojan Horse Defense



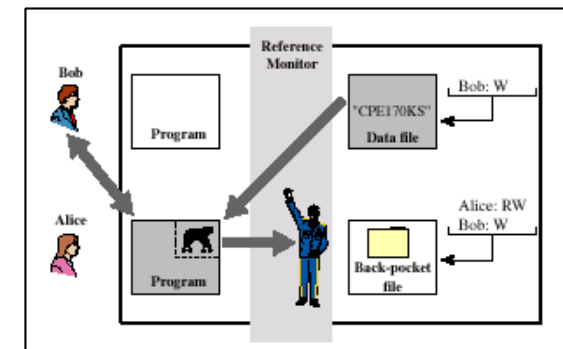
Alice induces Bob to invoke the trojan horse. Program detects it is being executed by Bob, reads the sensitive character string and writes it into Alice's back-pocket file

## Trojan Horse Defense



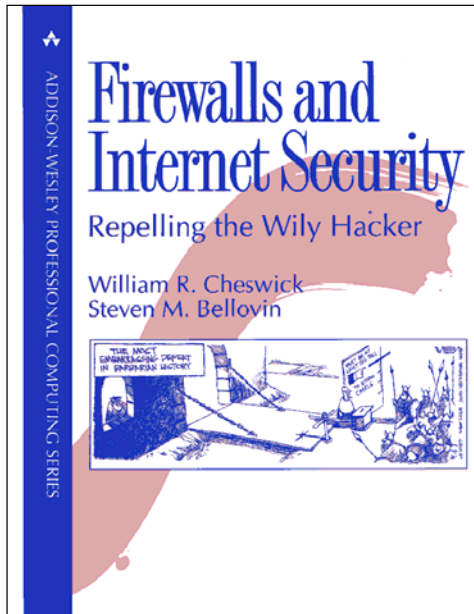
Two security levels are assigned, sensitive(higher) and public. Bob's stuff is sensitive and Alice's stuff is public.

## Trojan Horse Defense



If Bob invokes the trojan horse program, that program acquires Bob's security and is able to read the character string. However, when the program attempts to store the string, the no write down policy is invoked





Firewall

A classic in the field published in 1994. Know for its ⚡ “bombs” which indicated a serious risk

## Important URLs

- [Evolution of the Firewall Industry](#) - Discusses different architectures and their differences, how packets are processed, and provides a timeline of the evolution
- <http://csrc.nist.gov/publications/nistpubs/800-41/sp800-41.pdf> NIST Guidelines On Firewalls and Firewall Policy
- [Trusted Computing Group](#)  
Vendor group involved in developing and promoting trusted computer standards

Firewall