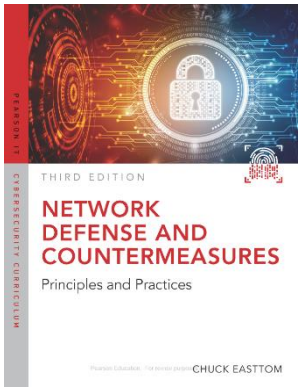
Fundamentals of Firewalls



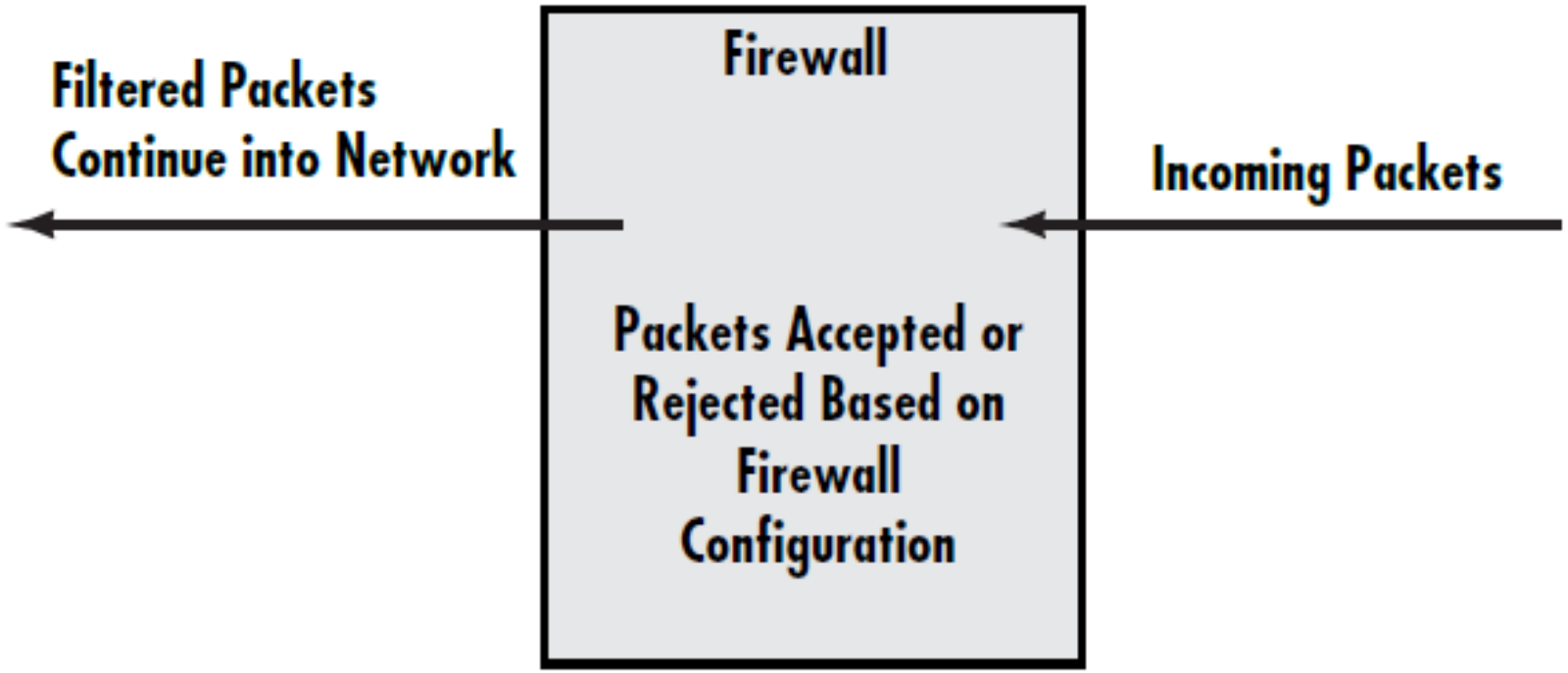
Based on slides accompanying the book
Network Defense and Countermeasures
by Chuck Easttom (2018)

Objectives

- Explain how firewalls work
- Evaluate firewall solutions
- Differentiate between packet filtering and stateful packet filtering
- Differentiate between application gateway and circuit gateway

What Is a Firewall?

- A barrier between the world and your network
- Provided via:
  - Packet filtering
  - Stateful packet filtering
  - User authentication
  - Client application authentication



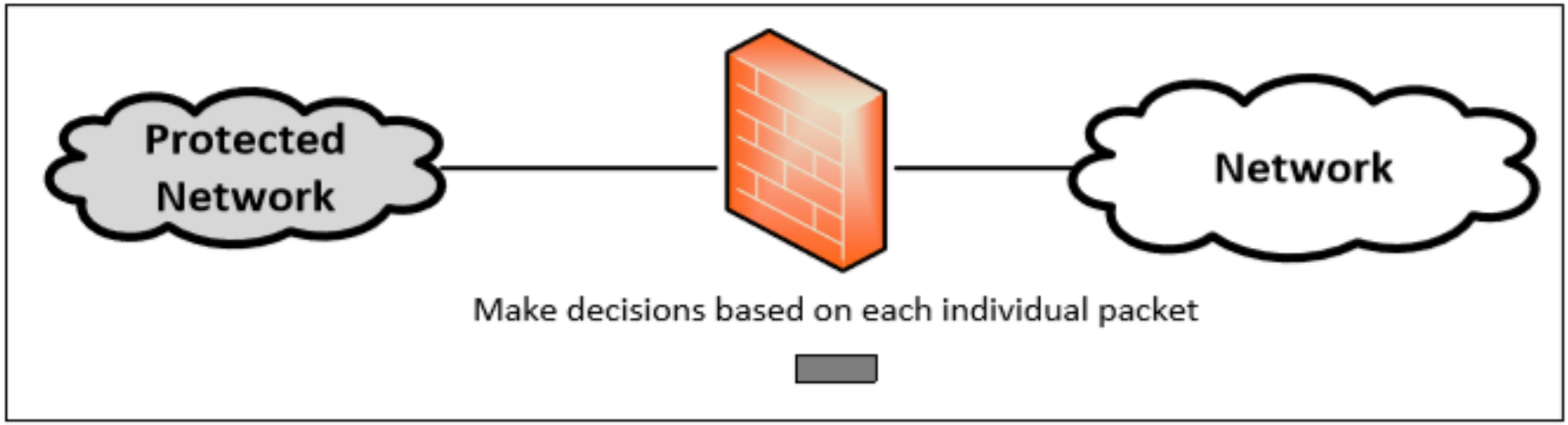
Types of Firewalls

- Network-level
  - Packet filter
  - Stateful packet filter
  - Circuit level gateway
- Application-level
  - Application gateway

Packet Filtering Firewall

- Very basic type of firewall
- Also referred to as “screening host” firewalls
- Works by examining a packet’s
  - Source address
  - Destination address
  - Source port
  - Destination port
  - Protocol type

Packet Filter Firewall



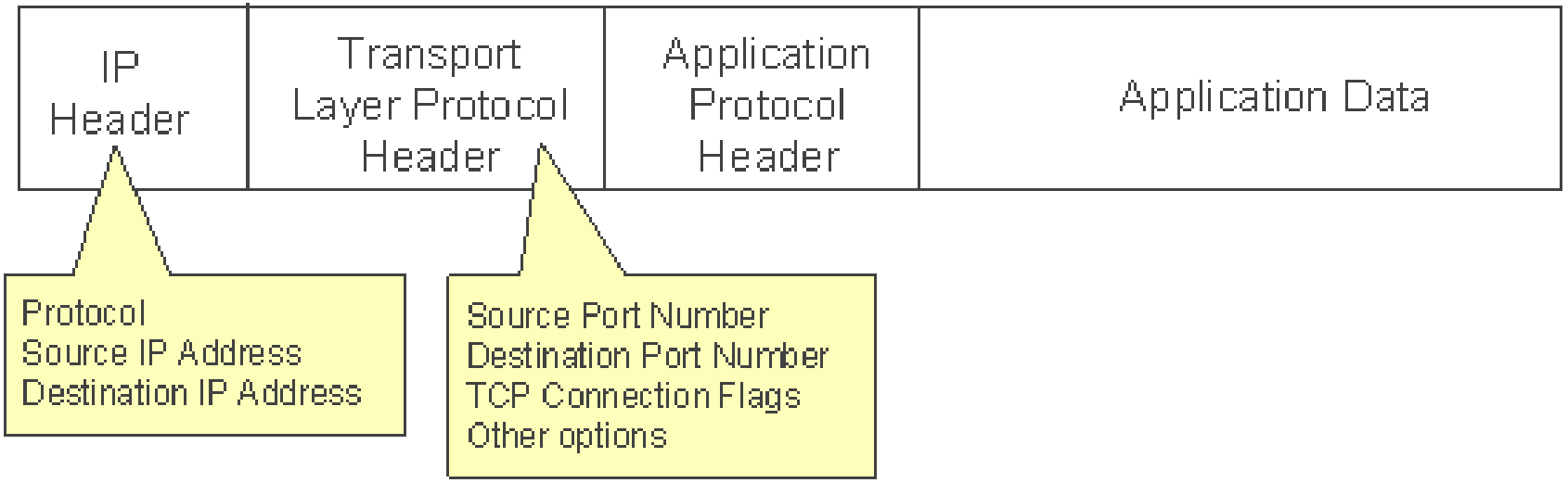
- Doesn’t pay attention to if the packet is a part of existing stream or traffic.
- Doesn’t maintain the states about packets. Also called Stateless Firewall.

- Controls traffic based on the information in packet headers, without looking into the payload that contains application data.

Proxy Servers and Firewalls

Static Packet Filtering 7/19

- A **router** is an internetworking device that transfers IP packets between two or more network segments (interfaces)
- Most routers can be used to screen and selectively filter IP packets (i.e., **screening routers**) based on the network interface and information that is found in the headers of the IP packets



Common Packet Filtering Products

- Firestarter – free Linux firewall
- Avast Internet Security - Windows only
- Zone Alarm Firewall
- Comodo Firewall

Packet Filtering Firewall Disadvantages

- *'Stateless'* (aka. *Static* packet filtering)
- Does not compare packets
- No authentication
- Susceptible to SYN and Ping flood attacks
- Does not track packets
- Does not look at the packet data, just the header
- Not necessarily the most secure firewall

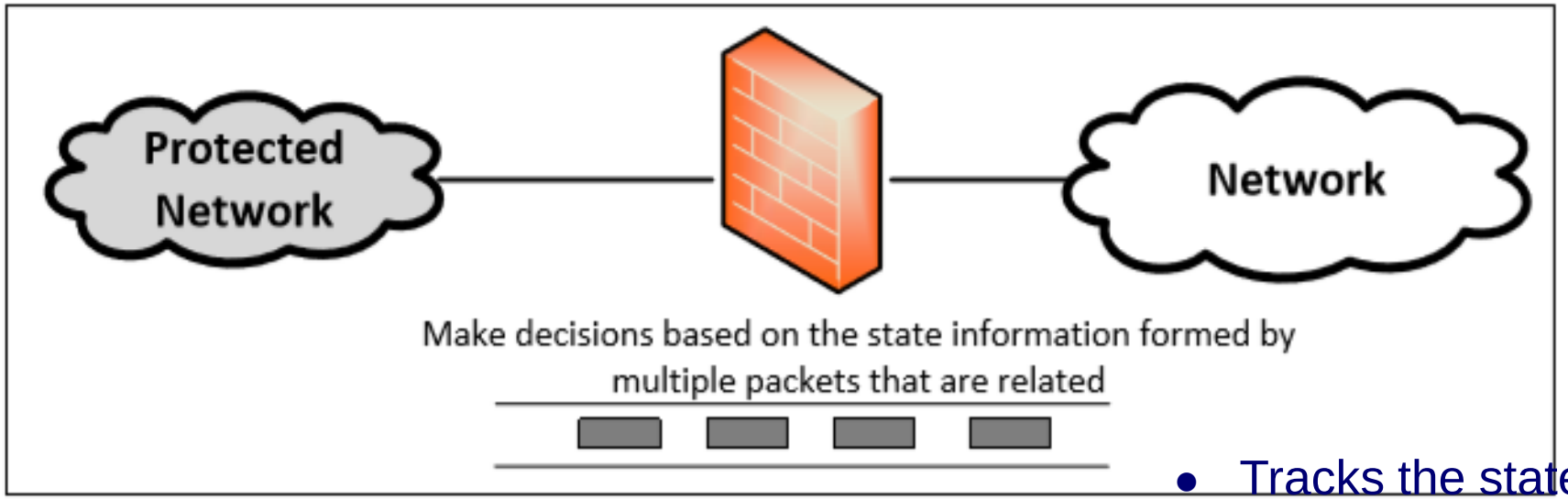
Packet Filtering Firewall Rules

- Rules should cover:
  - What types of protocols to allow
    - FTP
    - SMTP
    - POP3
  - What source ports to allow
  - What destination ports to allow
  - What source IP addresses to allow

Stateful Packet Inspection

- Aka. *Dynamic* packet filter
- Being aware of the **context** of packets makes them less susceptible to flood attacks
  - Knows if packet is part of a larger stream
  - Recognizes whether source IP is within the firewall
  - Can look at the **contents** of the packet
- When possible, the recommended firewall solution (over the stateless packet filtering)

Stateful Firewall



- Example : Connections are only allowed through the ports that hold open connections.

- Tracks the state of traffic by monitoring all the connection interactions until is closed.
- Connection state table is maintained to understand the context of packets.

Stateful / Dynamic Packet Filtering

- A *dynamic* packet filter maintains *state* information about past IP packets to make more intelligent decisions about the legitimacy of present and future IP packets
- State information are stored in an internal database
- Subsequent packets belonging to the same association can pass quickly through the stateful inspection device

Application Gateway

- A program that runs on a firewall

  aka application proxy or application-level proxy
- Examines the connection between the client and the server applications

  Q: stateless or stateful?
- Enables administrators to specify what applications are allowed
- **Client app authenticated** first, followed by **user authentication**
- Computers behind the firewall are protected.

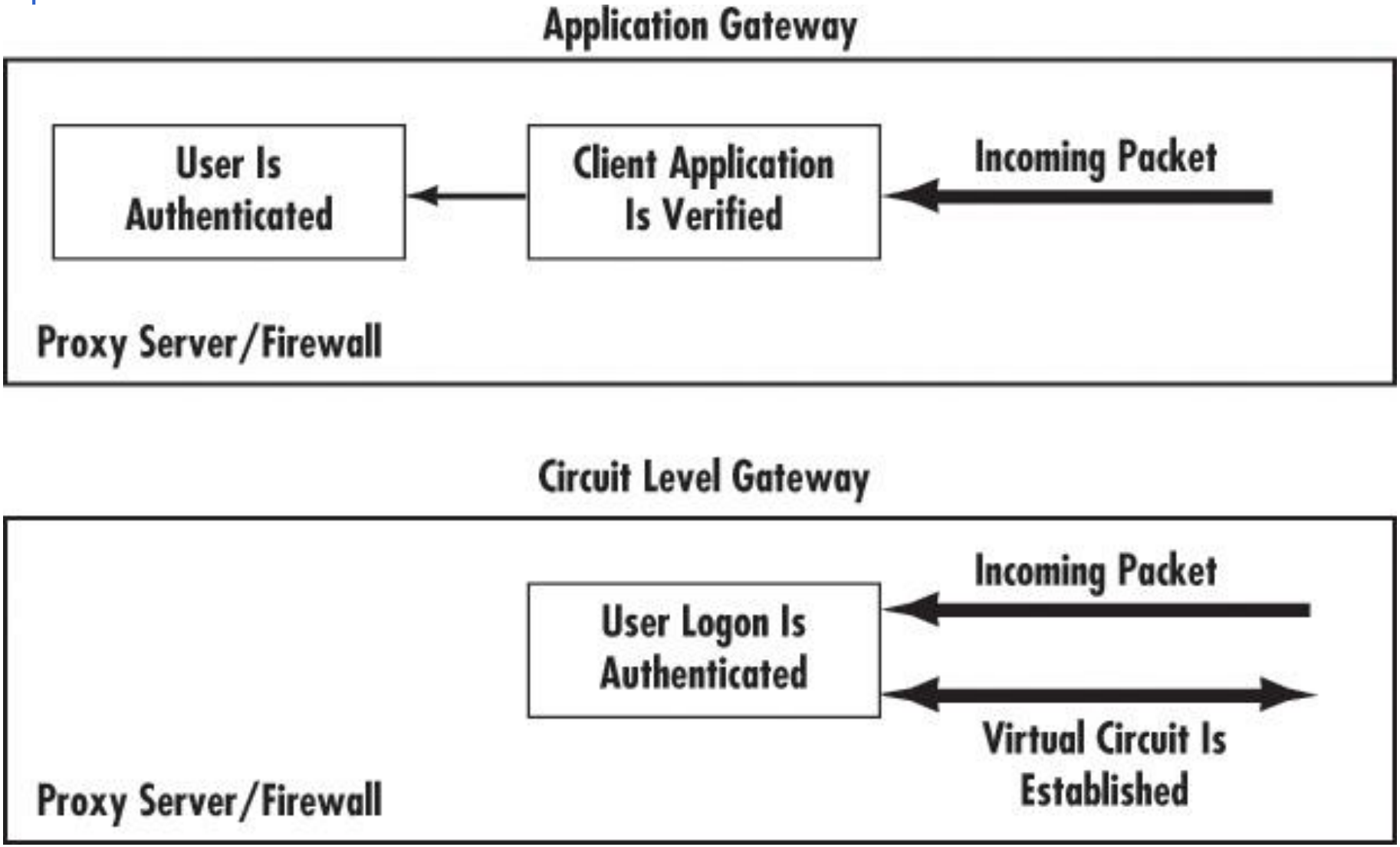


FIGURE 3-2 Application gateway vs. circuit level gateway

Application Gateway Disadvantages

- Disadvantages
  - Each app/protocol requires its own AG
  - Requires more system resources
  - Susceptible to flooding attacks (SYN and Ping)
    - Due to time it takes to authenticate user
    - When connection is made, packets are not checked, allowing a hacker to use an established connection to cause flooding

    --》 **mitigation?** User authentication?
- Product examples
  - Teros provides an AG for web servers
  - The Firebox from Watchguard Technologies

Circuit Level Gateway

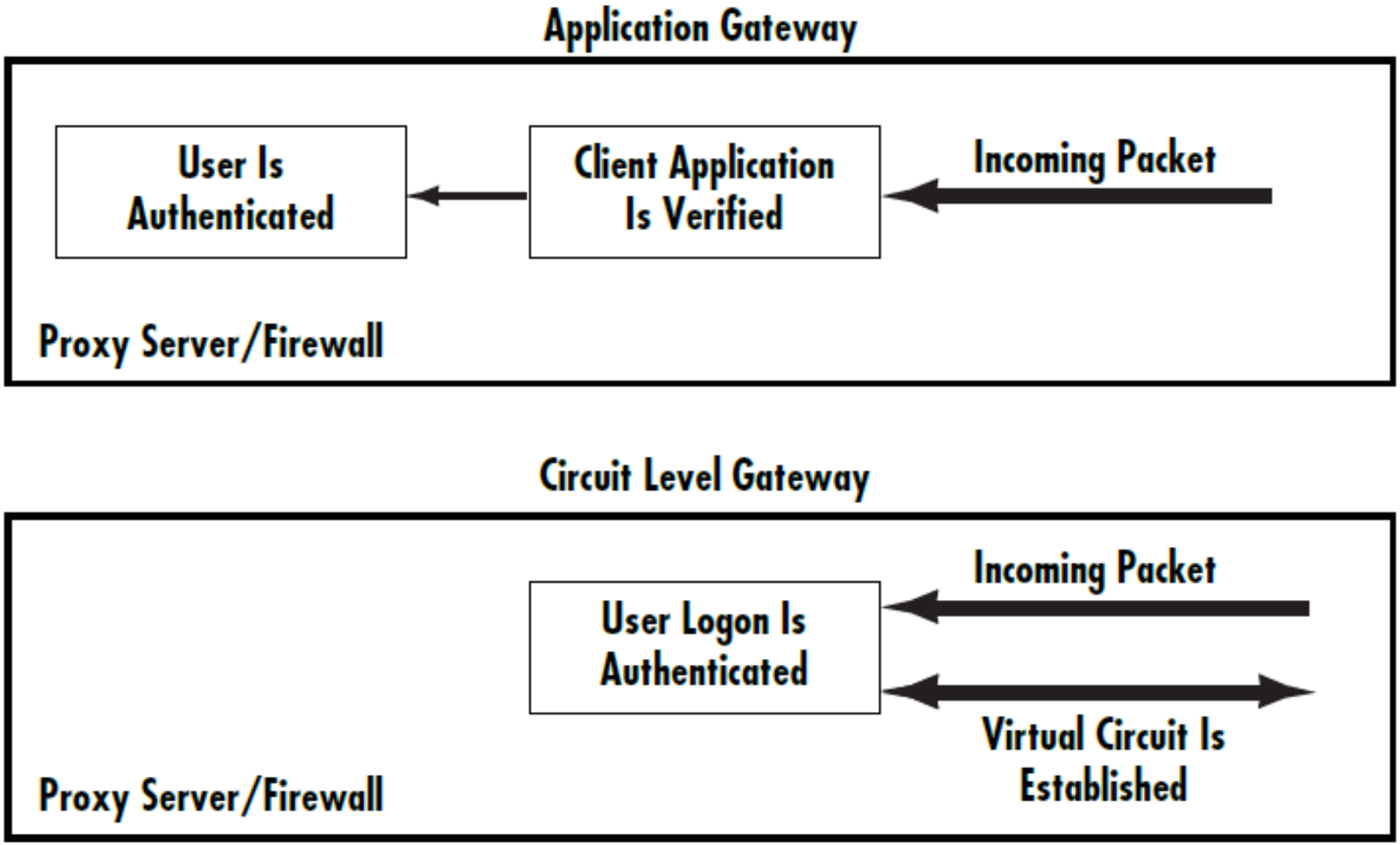
- More secure than application gateways
  - Authenticates the **user** first, before any further communication can take place
  - c.f., Application Gateway: **client app** is authenticated first, followed by **user** authentication

NOTE: The above notion is not shared by other sources. For example: https://www.open.edu/openlearn/science-maths-technology/computing-and-ict/systems-computer/network-security/content-section-9.6#

Q: Who performs the authentication?

- Virtual circuit is used to pass bytes between client and proxy server

Application vs. Circuit Level Gateway



Virtual circuits

- VC: Transportation of data over a packet switched network to simulate a dedicated physical layer link between the two ends
  - aka. virtual connection or virtual channel

(https://en.wikipedia.org/wiki/Virtual_circuit)

Virtual circuits vs Datagram networks

VC	- Connection-oriented - Reserved resources (cpu, memory buffers, network bandwidth) - A reserved path per circuit - Once the circuit is established, following packets are transmitted in order over that same circuit - do not need to be routed again	QoS • highly reliable • Packets do not need reordering at the receiving end
Datagrams **TCP**	- Connection-oriented protocol built on top of a connectionless protocol (e.g., IP) - No reserved resources - No reserved path	• Not as reliable as VC • But less costly

Virtual circuits vs Datagram networks

Datagrams **TCP**	- Connection-oriented protocol built on top of a connectionless protocol (e.g., IP) - No reserved resources as in VC - No reserved path	• Not as reliable as VC • But less costly
Datagrams **UDP**	- Connection-less protocol - No overhead for opening a connection, maintaining a connection, and terminating a connection	- efficient for broadcast and multicast type of network transmission

Circuit Level Gateway

- Typically implemented on high-end equipment (**NOTE:** This may not be true!)
- External users see only the proxy IP and not the internal client IP address
  - External systems do not see internal systems

  NOTE: This is also true in Application Level Gateways.
- May not work for some implementations (e.g., e-commerce)

Q: Which layer is the *circuit level gateway*?

Circuit-level Gateways/Firewalls

- A proxy server for TCP or UDP (at the transport layer)
- Goal: To allow a TCP/IP application to *traverse* (i.e., securely use) a firewall
- Is Located and running on a firewall
- Relays TCP connections:

  - They intercept TCP connection being made to a host behind them and complete the *handshake* on behalf of that host.

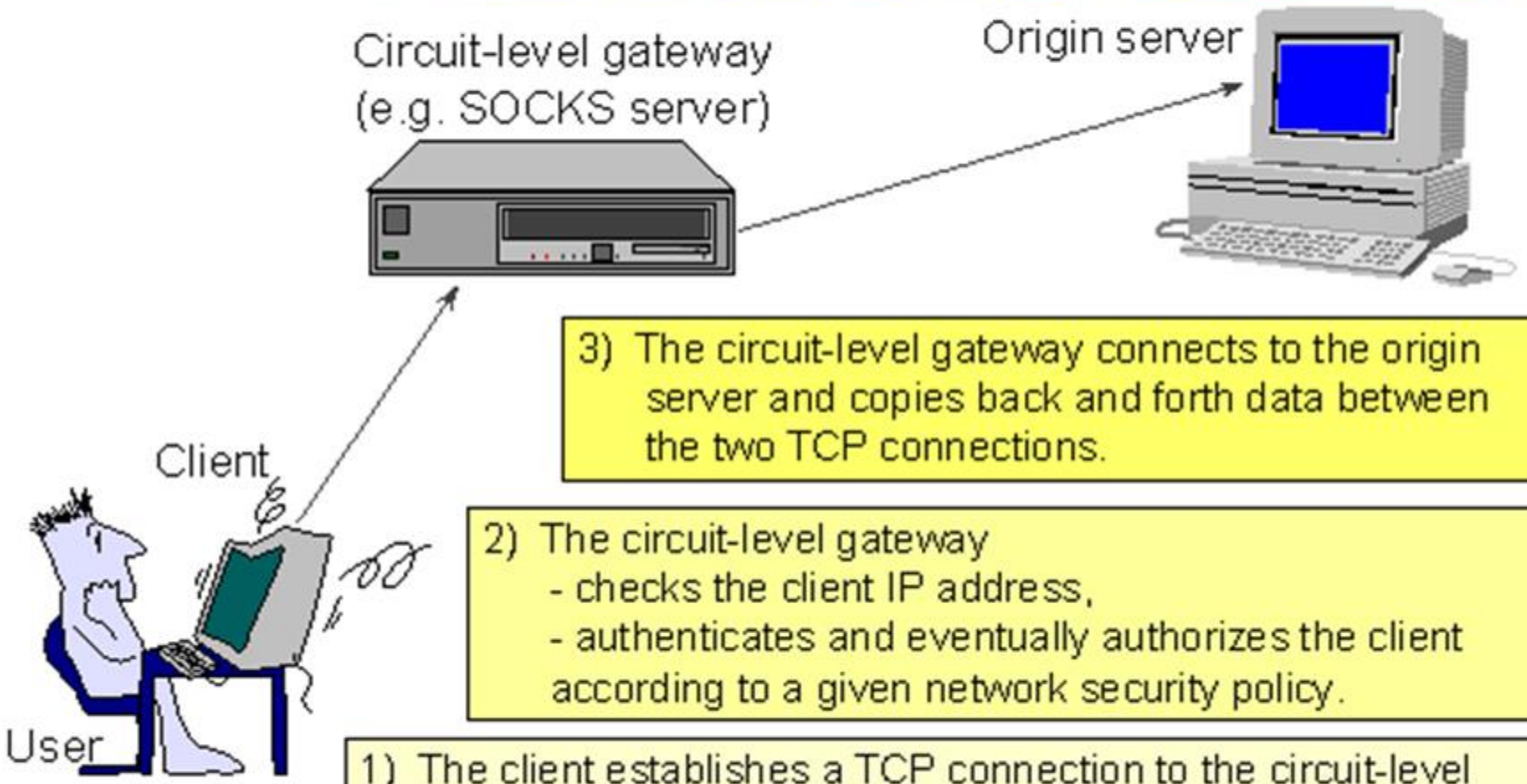
  - As soon as the connection is made, only data packets belonging to the connection are allowed to go through.
- It does not interfere with the data stream. ← Making it different from an *application-level gateway*
- Example: *SOCKS (*RFC1928
  SOCKS Protocol Version 5. *By M. Leech, M. Ganis, Y. Lee, R. Kuris, D. Koblas, L. Jones. March 1996)*

SOCKS

- The implementation of the SOCKS protocol typically involves the recompilation or relinking of TCP-based **client** applications to use the appropriate encapsulation routines in the SOCKS library. → *'socksified' clients*
- Procedure for TCP-based clients
  - When a TCP-based client wishes to establish a connection to an object that is reachable only via a firewall, it must open a TCP connection to the appropriate SOCKS port on the SOCKS server system. The SOCKS service is conventionally located on TCP port 1080.
  - If the connection request succeeds, the client enters a negotiation for the authentication method to be used, authenticates with the chosen method, then sends a relay request.
  - The SOCKS server evaluates the request, and either establishes the appropriate connection or denies it.

Proxy Servers and Firewalls

Circuit-level Gateways 11/19



3) The circuit-level gateway connects to the origin server and copies back and forth data between the two TCP connections.

2) The circuit-level gateway
- checks the client IP address,
- authenticates and eventually authorizes the client according to a given network security policy.

1) The client establishes a TCP connection to the circuit-level gateway and requests a second TCP connection to a remote server (origin server)

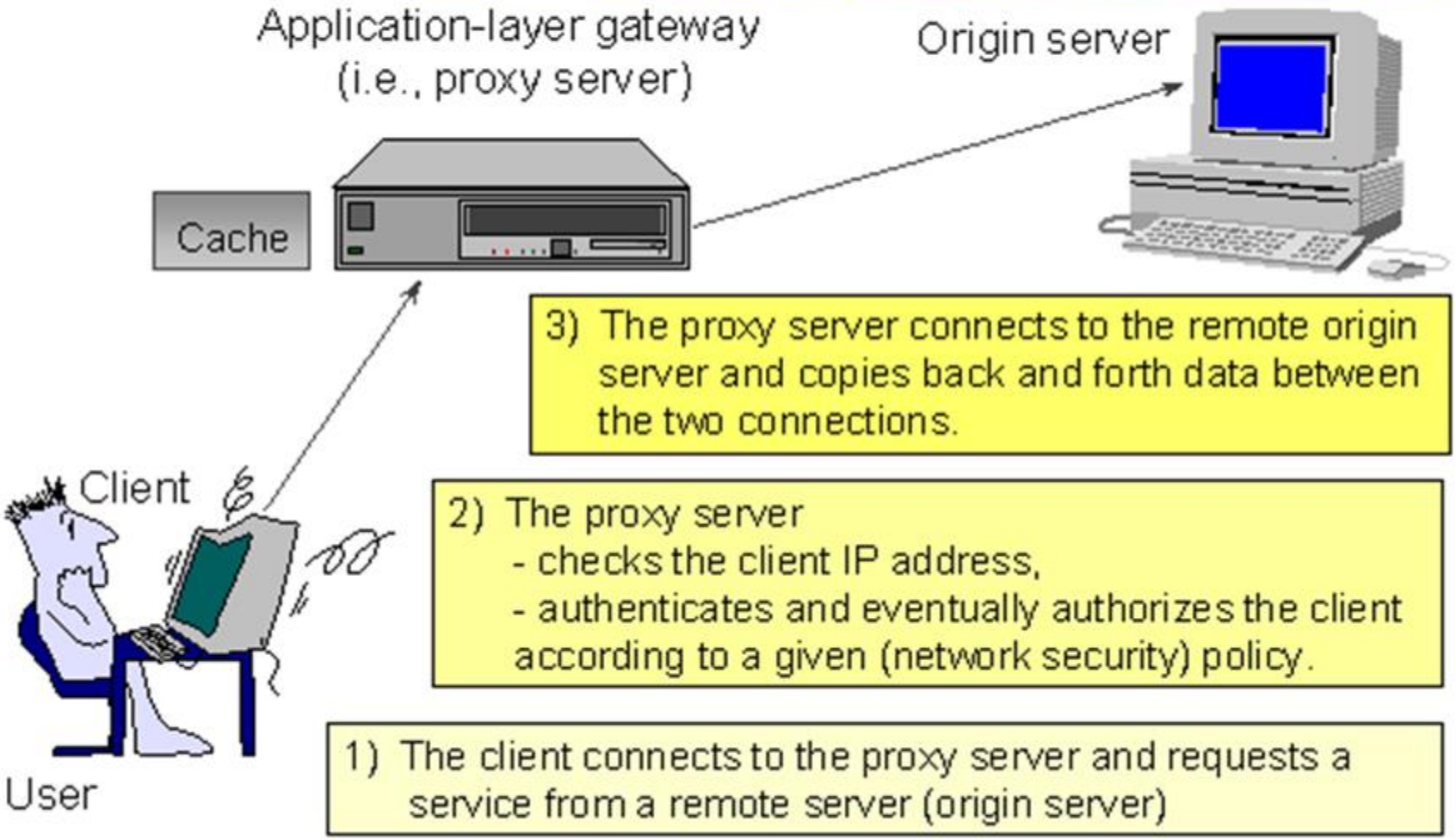
© 2003 eSECURITY Technologies Rolf Oppliger (www.esecurity.ch)

Application-level Gateways

- A proxy server that allows a specific application protocol to traverse a firewall.
- A sample scenario: The packet filter of a firewall blocks all inbound Telent and FTP sessions, unless the sessions are terminated by a bastion host.
  - Multiple application gateways may be running on the bastion host → a proxy server for FTP, a proxy server for Telent, …
  - A user who wishes to connect inbound to an intranet server must have his Telnet or FTP client connect to the application gateway.

Proxy Servers and Firewalls

Application-level gateways 12/19



3) The proxy server connects to the remote origin server and copies back and forth data between the two connections.

2) The proxy server
- checks the client IP address,
- authenticates and eventually authorizes the client according to a given (network security) policy.

1) The client connects to the proxy server and requests a service from a remote server (origin server)

© 2003 eSECURITY Technologies Rolf Oppliger (www.esecurity.ch)

Application-level Gateways

- To properly authenticate the user, an application gateway must have access to authentication and authorization information, either locally or remotely:
  - User-level authentication info may be stored locally on the firewall
  - User-level authentication info may be stored in a centralized authentication server (e.g., RADIUS, TACACS+)

Trade-offs of Firewalls

- **Advantages**:
  1. Provides basic access control services for an intranet
  2. Provides a centralized filtering/gateway function
  3. (To some degree) Relieves individual hosts the responsibility of having a filter or firewall itself
  4. Centralized management of filtering rules
- **Limitations**: next

Trade-offs of Firewalls

- Limitations:
  1. Cannot protect sites and corporate intranets against insider attacks → internal / intranet firewalls
  2. Can be circumvented by *tunneling* unauthorized application protocols in authorized ones
  3. Little protection against attacks embedded in the *data* field of a packet (e.g., virus-infected programs or data files, malicious Java applets, malicious ActiveX controls, …)
  4. May foster a false sense of security → lax security within the firewall perimeter