Firewalls are ~~Good~~ *Necessary*

Steven M. Bellovin

smb@research.att.com

908-582-5886

AT&T Bell Laboratories

Murray Hill, NJ 07974

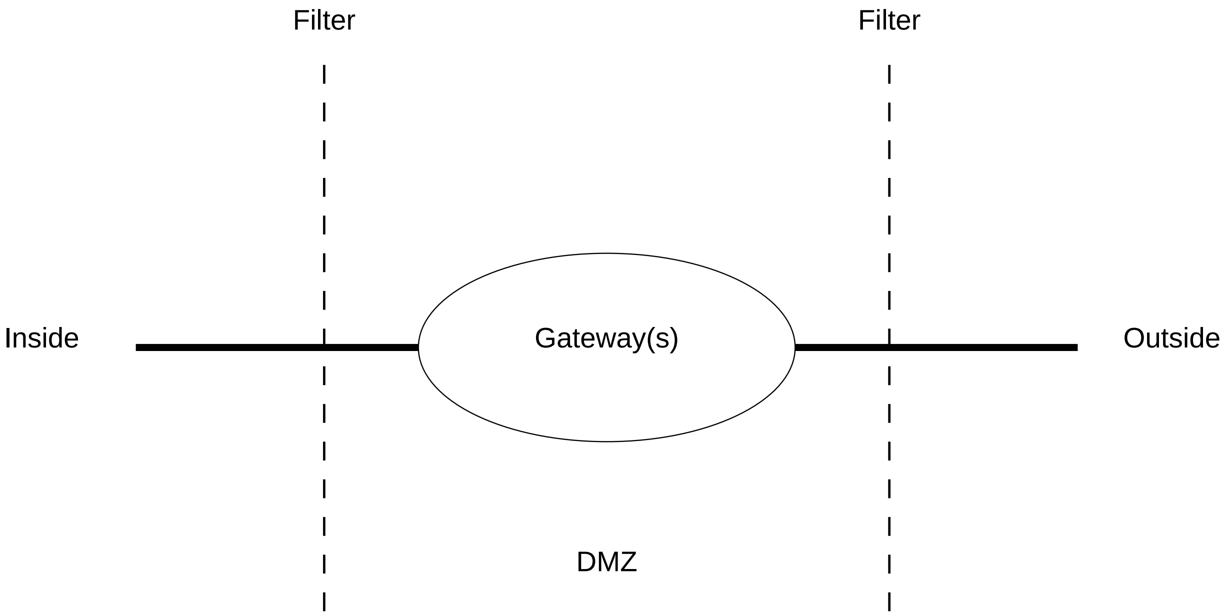
What's a Firewall

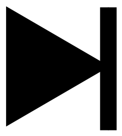
- Barrier between *us* and *them*.
- Limits communication to the outside world.

⇒ The outside world can be another part of the same company.

- Only a very few machines exposed to attack.

Schematic of a Firewall





Why Use Firewalls?

- Most hosts have security holes.

  Proof: Most software is buggy. Therefore, most security software has security bugs.

- Firewalls run much less code, and hence have few bugs (and holes).
- Firewalls can be professionally (and hence better) administered.
- Firewalls run less software, with more logging and monitoring.
- They enforce the partition of a network into separate security domains.
- *Without such a partition, a network acts as a giant virtual machine, with an unknown set of privileged and ordinary users.*

Should We Fix the Network Protocols?

- Network security is not the problem.
- Firewalls are *not* a solution to network problems. They are a network response to a host security problem.
- More precisely, they are a response to the dismal state of software engineering; taken as a whole, the profession does not know how to produce software that is secure, correct, and easy to administer.
- Consequently, better network protocols will not obviate the need for firewalls. The best cryptography in the world will not guard against buggy code.
- That said, we need to engineer—and deploy—better security protocols.

Firewall Advantages

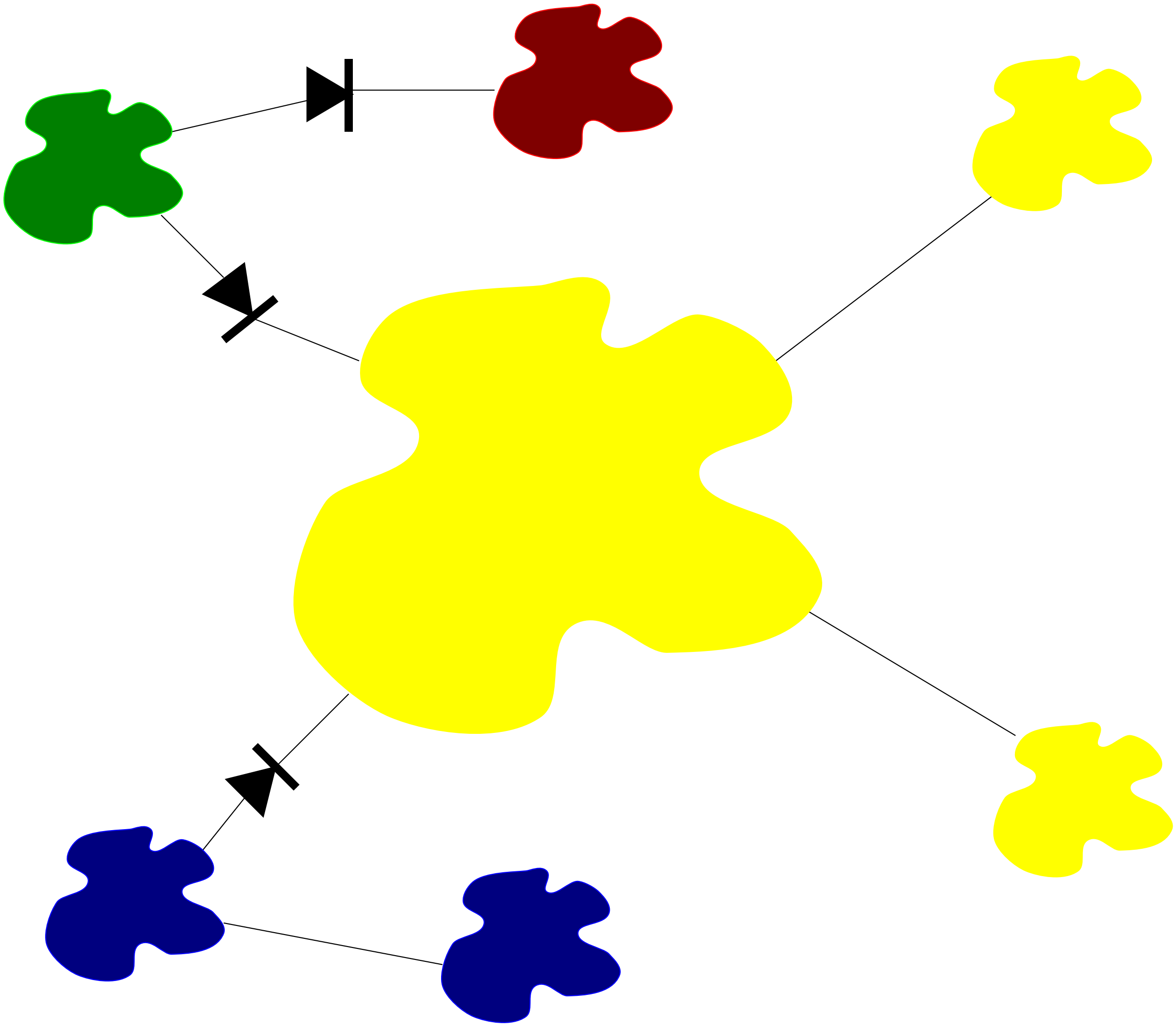
If you don't need it, get rid of it.

- No ordinary users, and hence no `/etc/passwd` entries.
- Run as few servers as possible (zap `rlogin`, `finger`, etc.)
- Install conservative software (eliminate `sendmail`, don't get the latest fancy `ftpd`, etc.)
- Log everything, and monitor the log files.
- Keep copious backups, including a "Day 0" backup.

Ordinary machines cannot be run that way.

Positioning Firewalls

Firewalls protect *administrative* divisions.



Types of Firewalls

- Packet Filters
- Application Gateways
- Circuit Relays

Many firewalls are combinations of these types.

Packet Filters

- Router-based (and hence cheap).
- Individual packets are accepted or rejected; no context is used.
- Filter rules are hard to set up; the primitives are often inadequate, and different rules can interact.
- Packet filters a poor fit for `ftp` and `X11`.
- Hard to manage access to RPC-based services.

Sample Rule Set

$$\begin{array}{rrcl}
\textbf{block:} & \mathit{theirhost} & = & \textsc{spigot} \\
\textbf{allow:} & \mathit{theirhost} & = & \textit{any}\ \textbf{and} \\
 & \mathit{theirport} & = & \textit{any}\ \textbf{and} \\
 & \mathit{ourhost} & = & \textsc{our-gw}\ \textbf{and} \\
 & \mathit{ourport} & = & 25.
\end{array}$$

Incorrect Rule Set

$$\begin{array}{rrcl}
\textbf{allow:} & theirhost & = & \textit{any}\ \textbf{and} \\
 & theirport & = & 25\ \textbf{and} \\
 & ourhost & = & \textit{any}\ \textbf{and} \\
 & ourport & = & \textit{any}.
\end{array}$$

Any remote process on port 25 can call in.

The Right Choice

$$\begin{array}{lrcl}
\textbf{allow:} & \mathit{theirhost} & = & \textsf{\textit{any}}\ \textbf{and} \\
 & \mathit{theirport} & = & 25\ \textbf{and} \\
 & \mathit{ourhost} & = & \textsf{\textit{any}}\ \textbf{and} \\
 & \mathit{ourport} & = & \textsf{\textit{any}}\ \textbf{and} \\
 & (\mathit{bitset}(\texttt{ACK}) & \textbf{or} & \mathit{source} = \textrm{INSIDE}).
\end{array}$$

Permit *outgoing* calls.

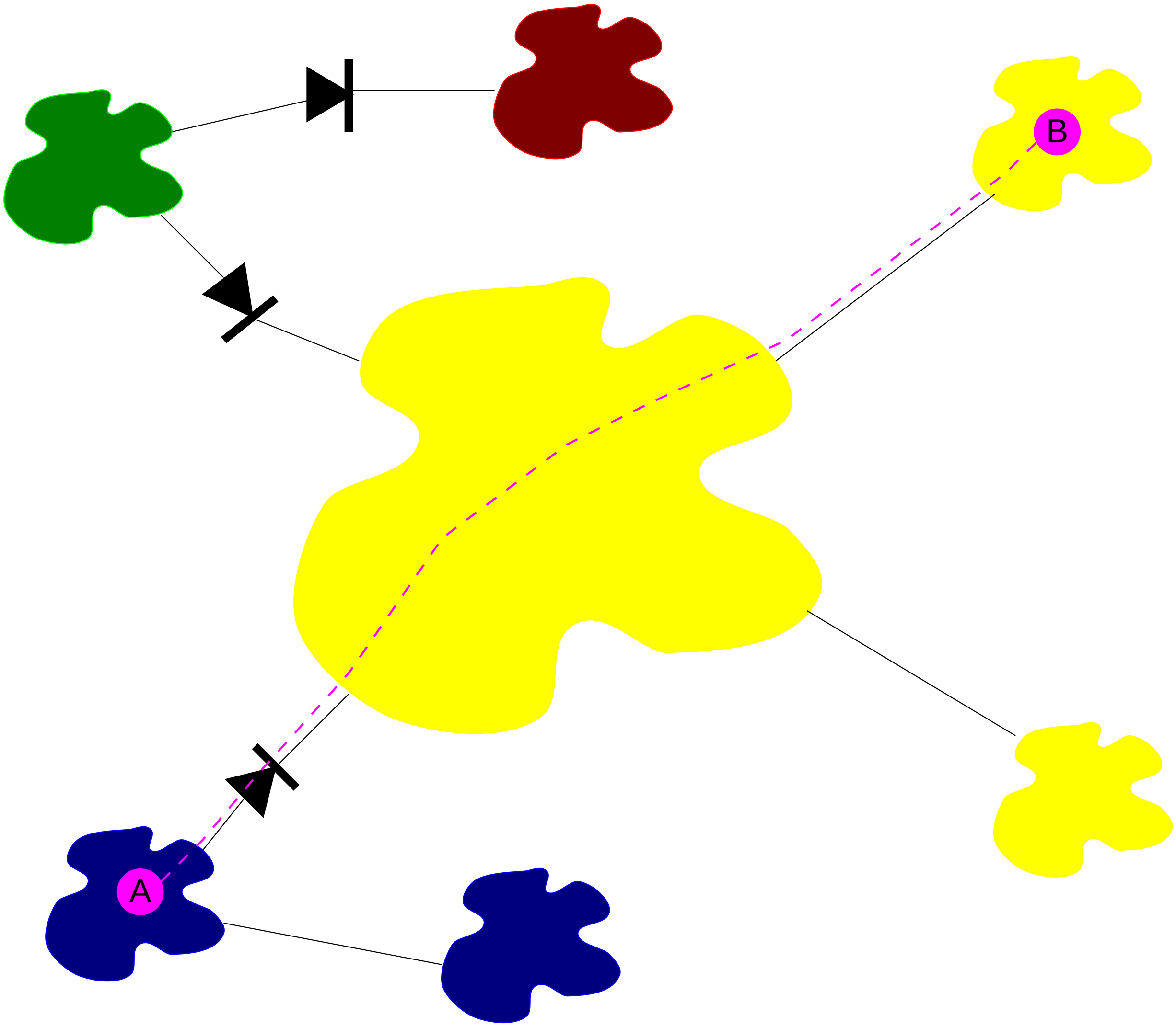
Application Gateways

- Gateway machine has custom program for each application.
- Facilities sometimes needed anyway (i.e., mail gateways).
- A good choice for X11 relays or for controlling outbound traffic.

Circuit Relays

- Messages are passed at the TCP level.
- No semantic processing by the gateway.
- Applications must be converted (but this isn't hard).
- More flexible than application gateway, but can be subverted.

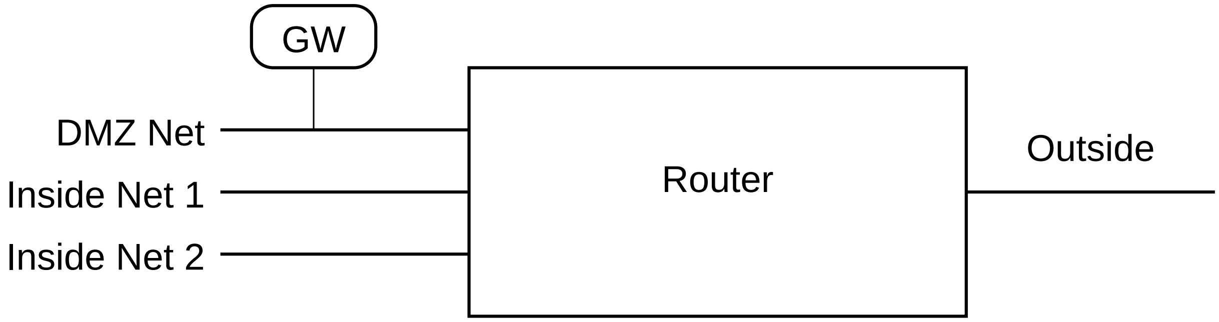
Creating Tunnels



But tunnels are often useful, especially if cryptographically protected.

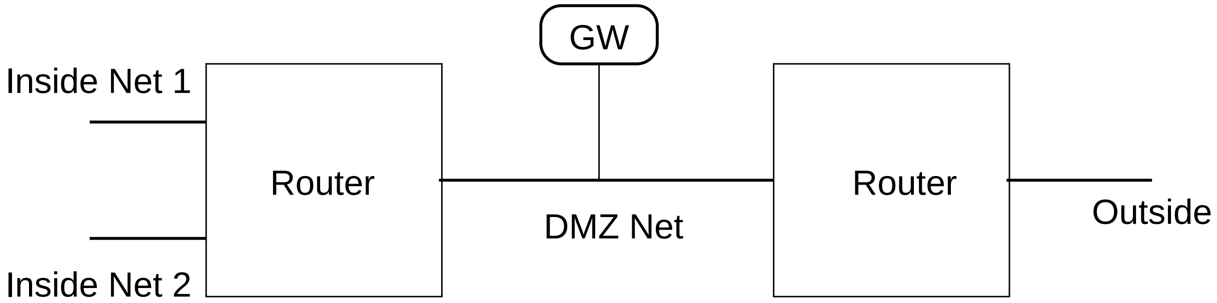
Single-Router Firewall

The cheapest design, but insecure with some brands of router.



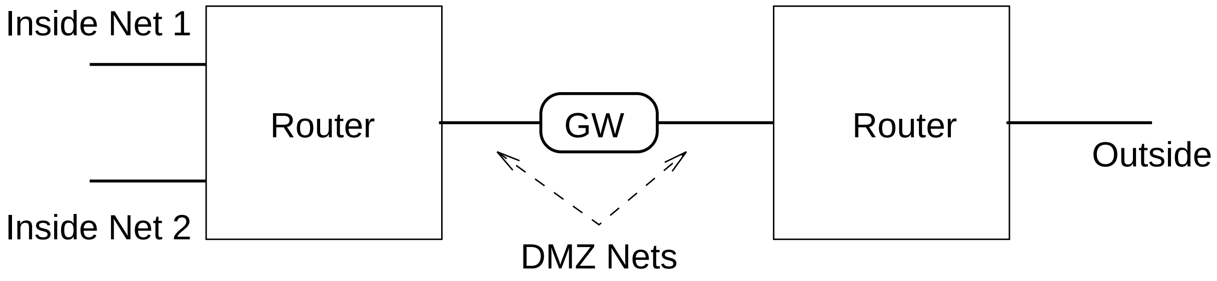
Double-Router Firewall

More secure, but more expensive.



“Belt and Suspenders”

A paranoid solution; the attacker has to go through the gateway, too.



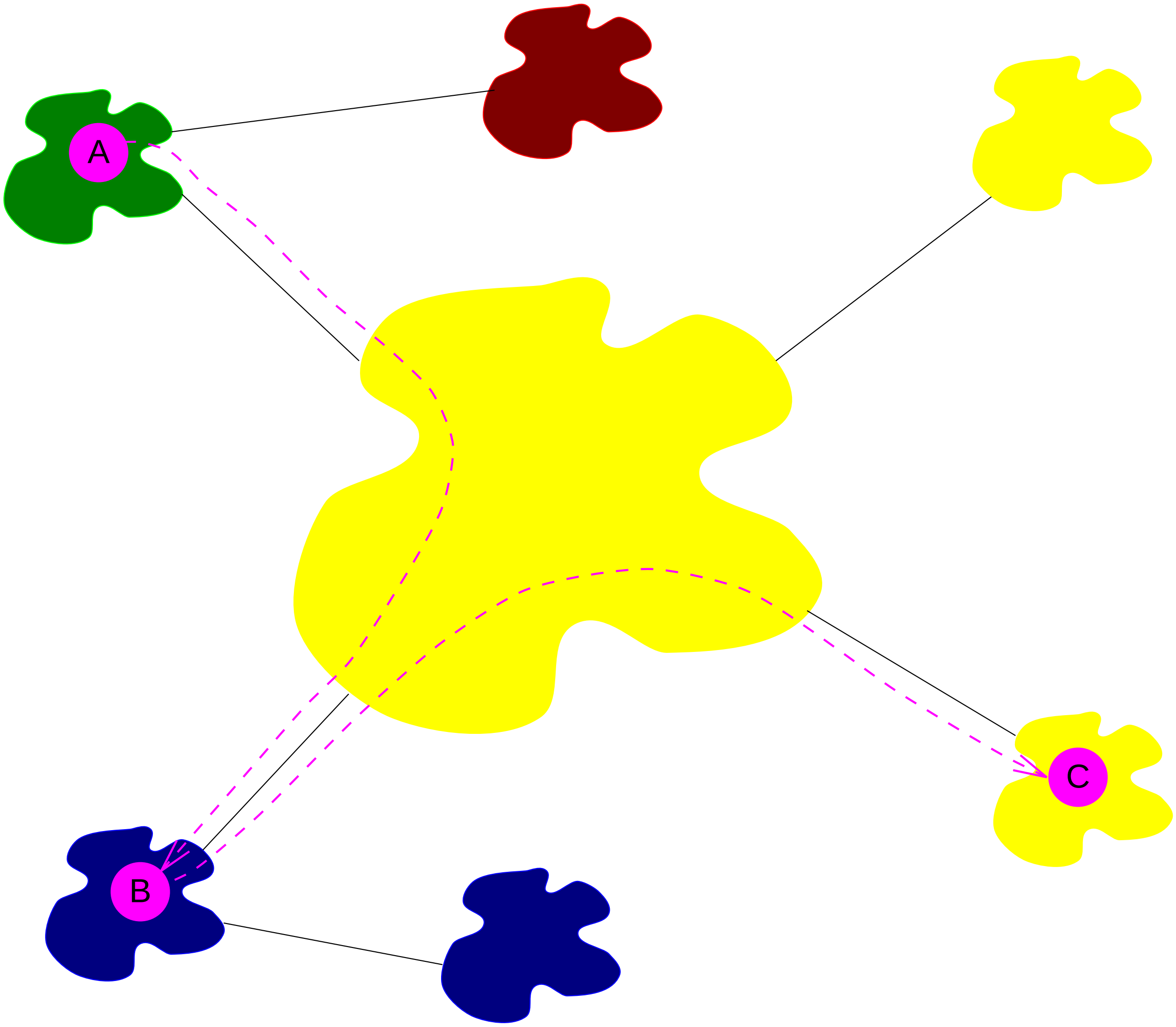
Providing Inbound Services

- Must allow some incoming traffic (mail, `ftp`, login, etc.)
- When possible, provide service on gateway machine (i.e., `ftp` repository).
- Use application gateway for pass-through services.
- High security, such as smart card authentication, desirable.

How Break-ins Can Spread

- Inappropriate `.rhosts` files.
- Logins via cracked passwords.
- Booby-trapped `telnet` commands.

Transitive Trust



If A trusts B and B trusts C, then A trusts C, whether it knows it or not.

Living With Firewalls

- Decide on a security policy.
- Decide which services fit that policy.
- Build/configure/tweak your firewall to permit those services.
- Evaluate new services using the same criteria.
- Block all others.

HACKERS