

Figure 2.1 Software and hardware service layers in distributed systems

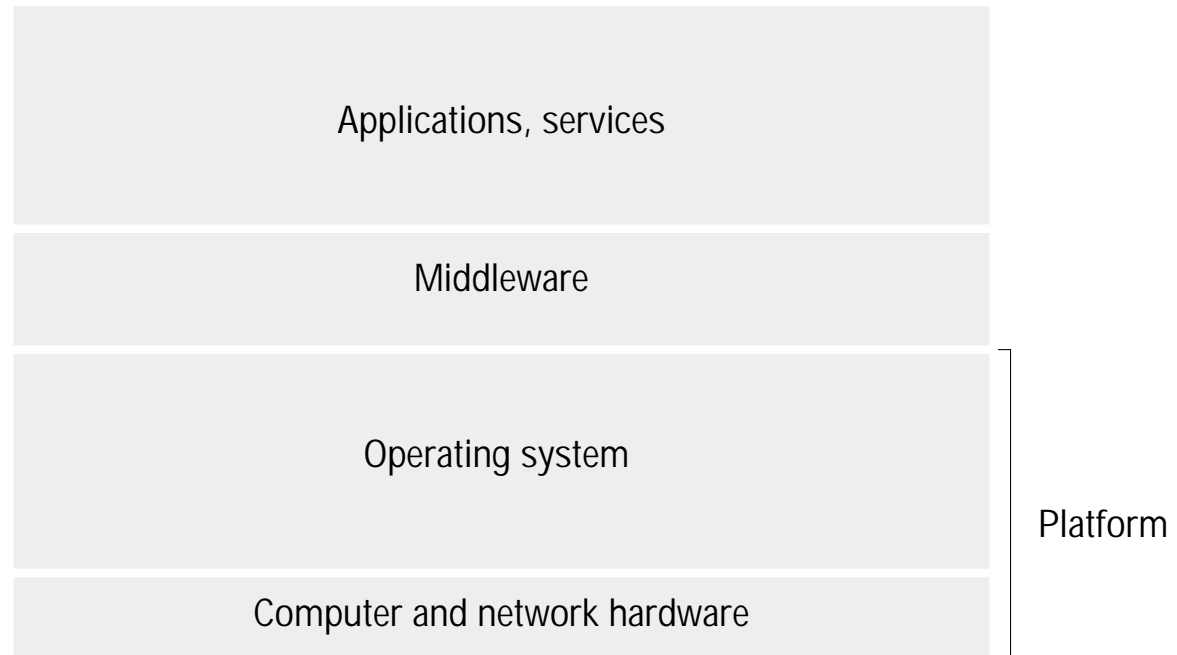


Figure 2.2 Clients invoke individual servers

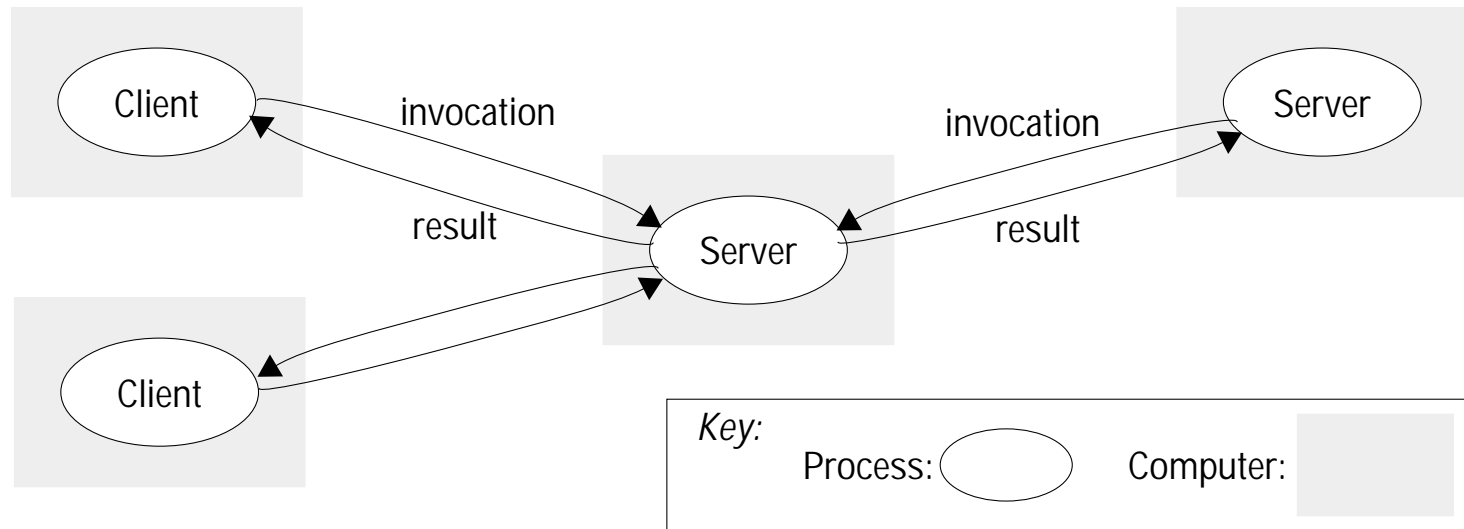


Figure 2.3 A service provided by multiple servers

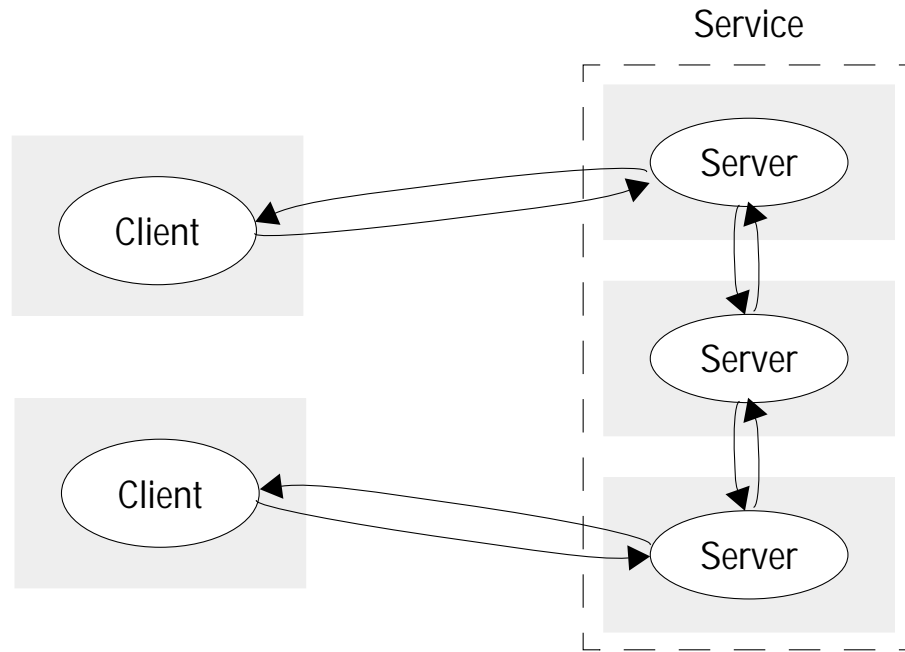


Figure 2.4 Web proxy server

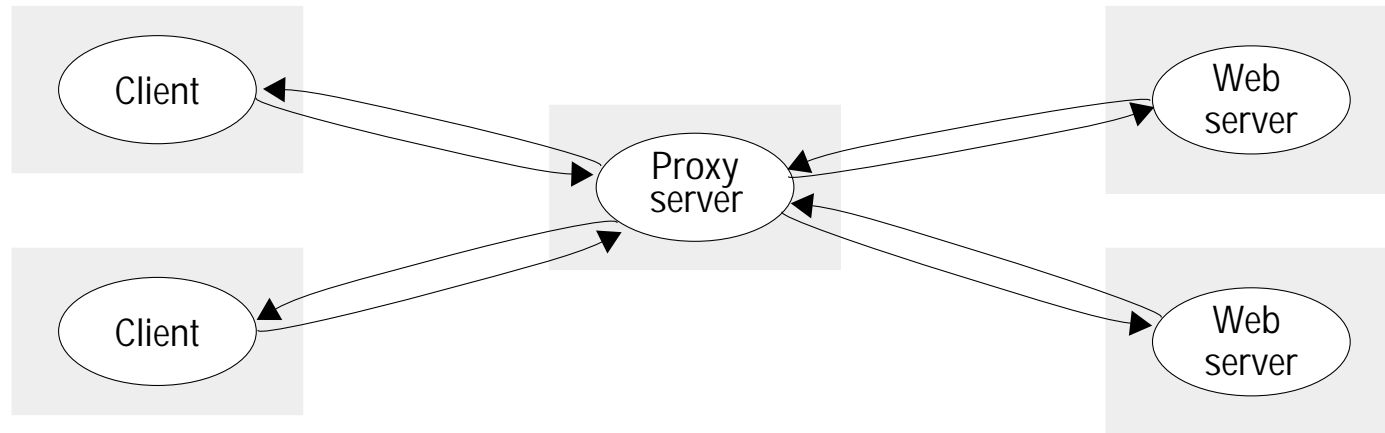


Figure 2.5 A distributed application based on peer processes

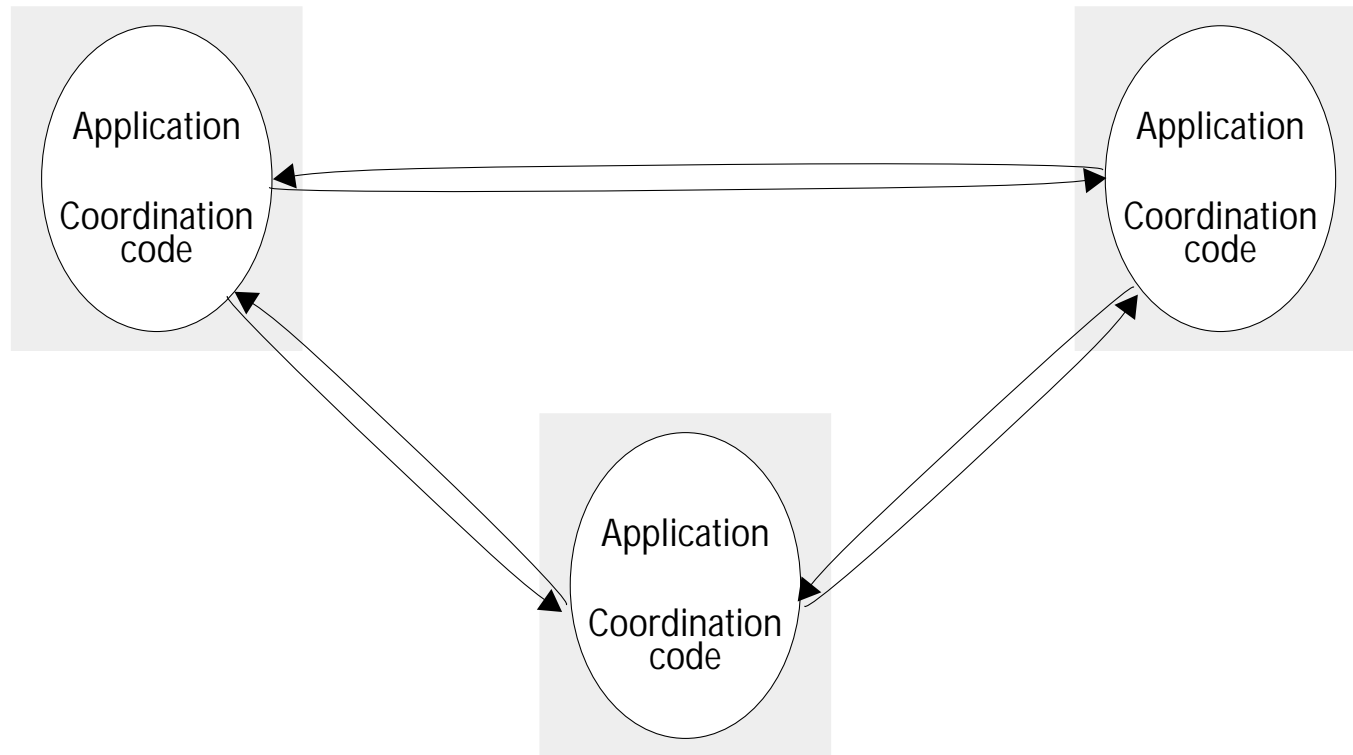
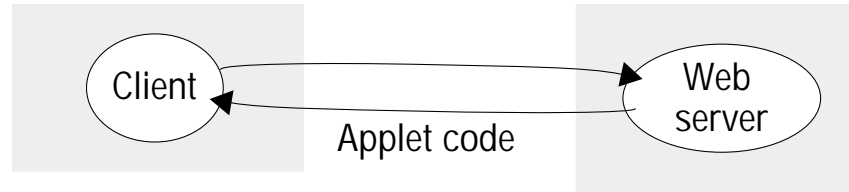


Figure 2.6 Web applets

a) client request results in the downloading of applet code



b) client interacts with the applet



Figure 2.7 Thin clients and compute servers

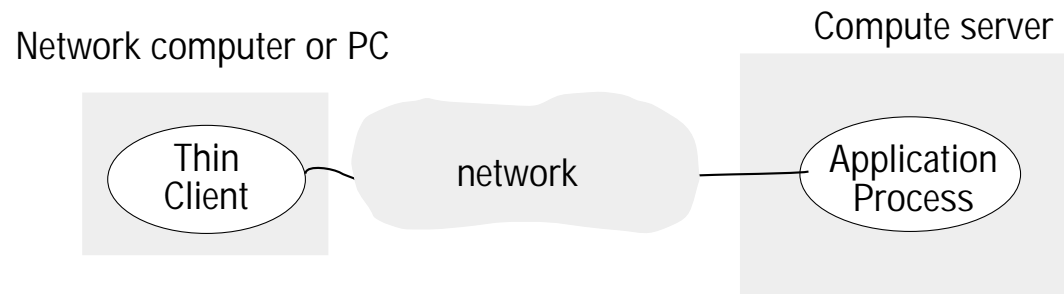


Figure 2.8 Spontaneous networking in a hotel

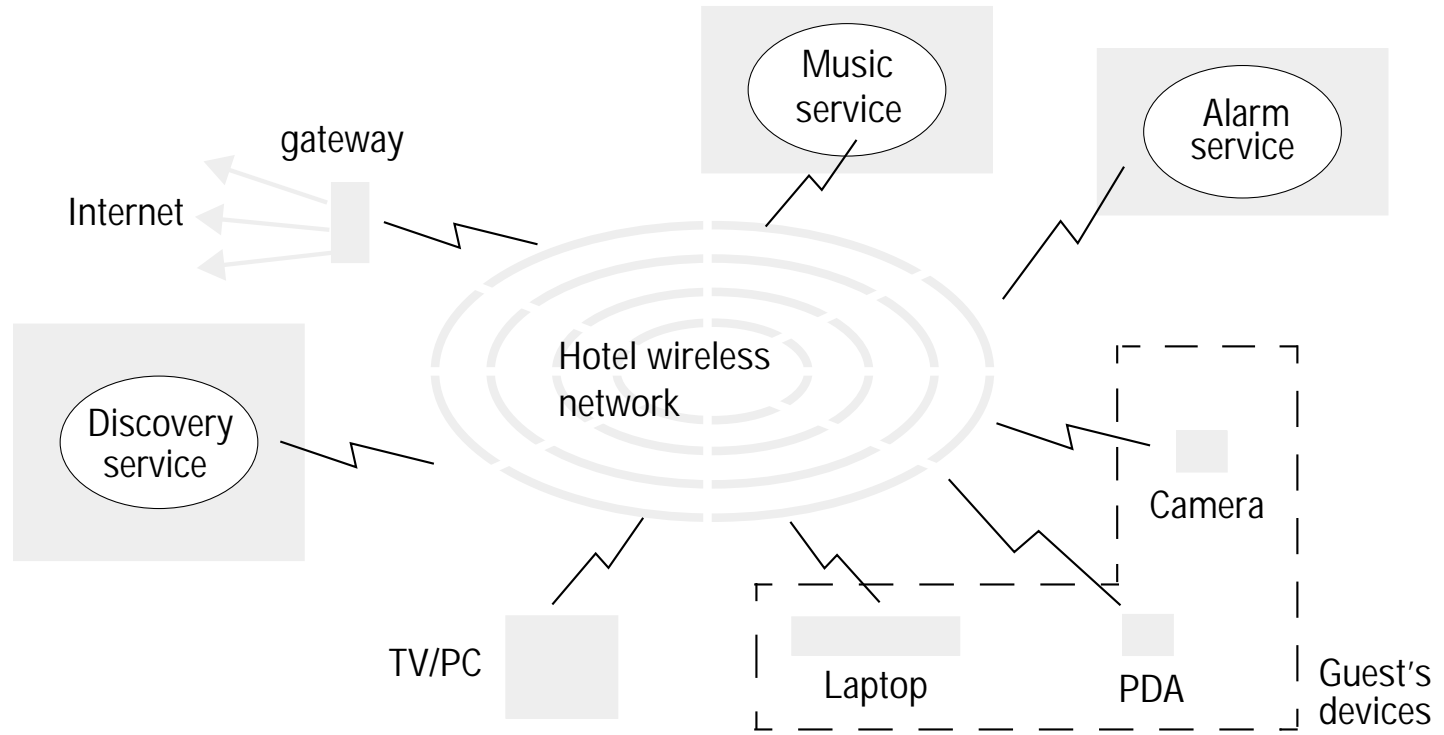


Figure 2.9 Real-time ordering of event.

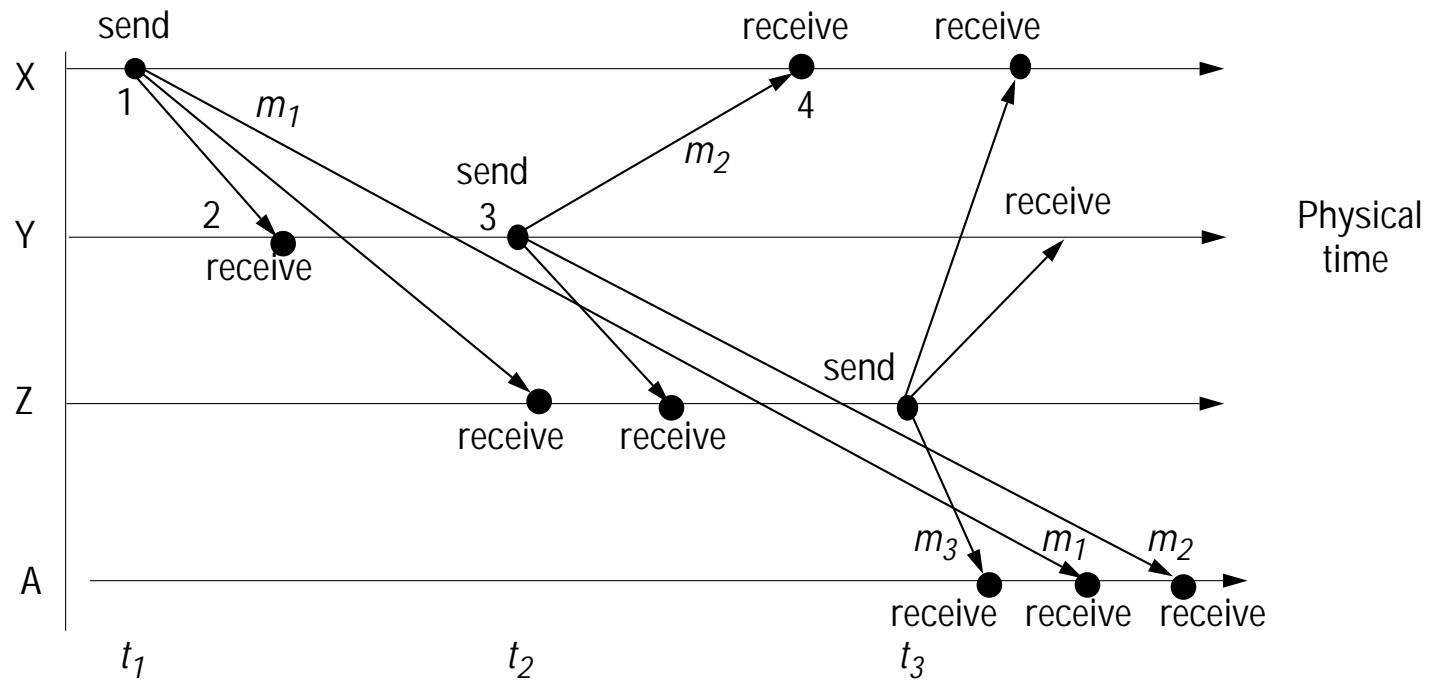


Figure 2.10 Processes and channels

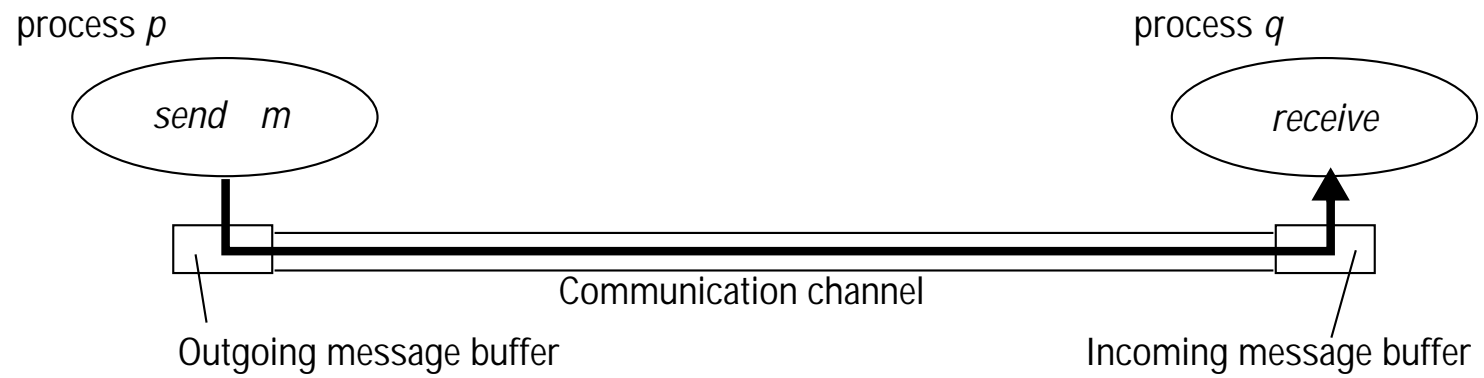


Figure 2.11 Omission and arbitrary failures

<i>Class of failure</i>	<i>Affects</i>	<i>Description</i>
Fail-stop	Process	Process halts and remains halted. Other processes may detect this state.
Crash	Process	Process halts and remains halted. Other processes may not be able to detect this state.
Omission	Channel	A message inserted in an outgoing message buffer never arrives at the other end's incoming message buffer.
Send-omission	Process	A process completes a <i>send</i> , but the message is not put in its outgoing message buffer.
Receive-omission	Process	A message is put in a process's incoming message buffer, but that process does not receive it.
Arbitrary (Byzantine)	Process or channel	Process/channel exhibits arbitrary behaviour: it may send/transmit arbitrary messages at arbitrary times, commit omissions; a process may stop or take an incorrect step.

Figure 2.12 Timing failures

<i>Class of Failure</i>	<i>Affects</i>	<i>Description</i>
Clock	Process	Process's local clock exceeds the bounds on its rate of drift from real time.
Performance	Process	Process exceeds the bounds on the interval between two steps.
Performance	Channel	A message's transmission takes longer than the stated bound.

Figure 2.13 Objects and principals

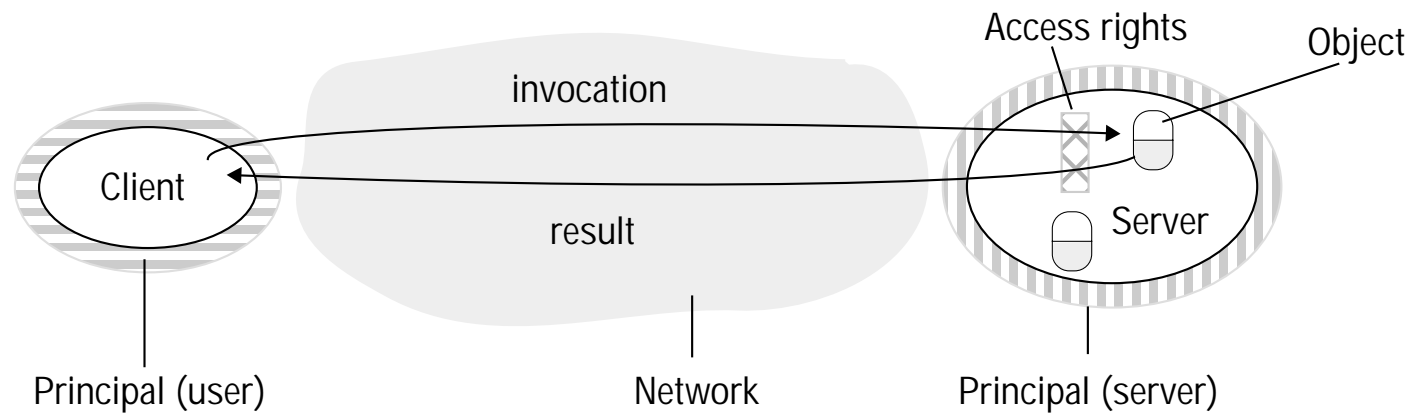


Figure 2.14 The enemy

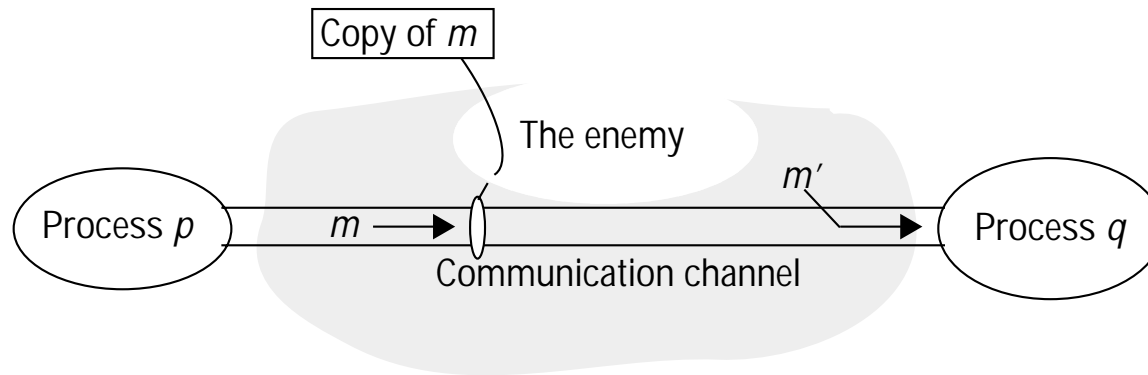


Figure 2.15 Secure channels

