In particular, `AuditPolicy` allows administrators to determine the channel that is used for logging security event occurrences. `AuditPolicy` further defines operation `set_ audit_selectors`, which can be used to determine the types of events for which an entry should be written into the audit channel. For example, by passing a list including event types `AuditPrincipalAuth` and `AuditSessionAuth` only authentication events will be logged. Finally, the `selectors` parameter allows administrators to determine which auditing information is to be written for each event.

Summary

In this section, we have seen an example of how object-oriented middleware supports the higher-level security concepts that we have introduced in previous sections. We have discussed how the CORBA Security service supports authentication, access control, non-repudiation and auditing. As with other CORBA services, the security service standardizes the interface of objects that are needed for security management purposes. We have seen that client programmers, server programmers and administrators need different interfaces for controlling the security of a distributed object-based system.

Key Points

- Distributed objects have to rely on networks to be able to communicate with each other. The message traffic on these networks can be eavesdropped on and tampered with by non-authorized third parties. The construction of trustworthy distributed object systems therefore demands that message traffic is secured against such attacks and that infiltration of distributed object systems by attackers is prevented.
- Encryption techniques are used to prevent eavesdropping and message tampering. Public or secret key methods are used to encrypt a marshalled object request before it is transmitted using an insecure network.
- The distribution of public or secret keys is achieved by a trusted key distribution service which implements a key distribution protocol. The Needham/Schroeder protocol is such a protocol.
- Encryption is also used for authentication. Authentication establishes the evidence that principals are who they claim to be. It can be achieved by demonstrating that the principal possesses a key, which can then be seen as evidence that the principal is authorized to perform certain operations.
- Access control is based on authentication. Middleware associate credentials with principals during authentication. These credentials determine the rights or privileges that have been granted to the principal to use the distributed object system. Administrators then assign the required rights to operations of objects or object types and the access control of the middleware ensures that only those principals who possess sufficient access rights can execute these operations.
- Non-repudiation is concerned with the generation of irrefutable evidence that principals have requested certain operations or that servers have received these