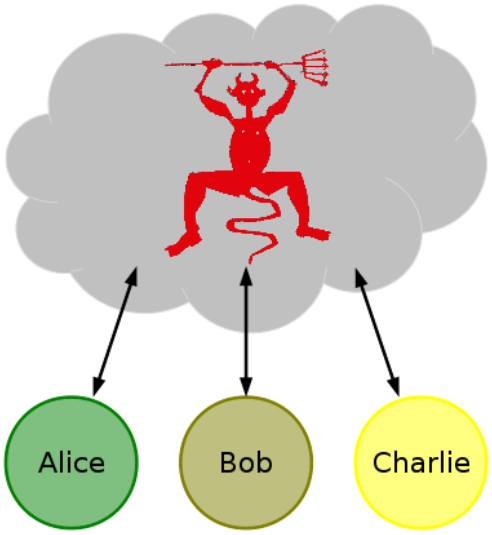
1 Introduction



Goals

- Explain principles behind reliable and secure distributed systems.
- Exploit replication as the primary means to tolerate faults.
- Describe some real-world applications in cluster computing and cloud computing.

Overview of topics

1. Dependability
2. Reliable broadcast
3. Distributed storage
4. Consensus
5. System examples
6. Distributed cryptography and proactive recovery
7. Integrity and confidentiality for data stored by untrusted servers
8. Confidentiality for computation on untrusted servers

Models

Components: processes/parties, network channels, message passing.

Time: synchronous, partially synchronous, asynchronous.

Channels: point-to-point, broadcast, reliable, authenticated, secret.

Failures: fail-stop, crash, omission, crash/recovery, arbitrary (Byzantine).

Formal models: I/O automata, non-determinism, unbounded time (for distributed systems), Turing machines, polynomially-bounded time (for cryptography).

Techniques

Distributed communication: reliable broadcast, causality, view-synchrony, view-based group communication; consensus, impossibility of asynchronous consensus; failure detectors; consensus, atomic broadcast, atomic commit; service replication.

Distributed cryptography: secret sharing; threshold cryptography, threshold encryption, threshold signatures, and threshold pseudorandomness; proactive security.

Service replication: primary-backup, state machine-replication, atomic broadcast.

Data replication: quorums, distributed storage, erasure coding.

Applications

Cluster computing: Group communication, Yahoo!'s ZooKeeper group synchonization service.

Distributed services: Cloud platforms, Domain Name System (DNS) security.

Distributed storage: Amazon's Dynamo, Windows Azure storage, IBM's GPFS distributed file system.

Literature

References are posted on the course website:

`http://www.zurich.ibm.com/~cca/sft13/`