Scaling Security for Big, Parallel File Systems

Andrew Leung and Ethan Miller
University of California, Santa Cruz
{aleung, elm}@cs.ucsc.edu
FAST 2007 Work-in-Progress

Motivation

- Large systems hard to secure
  - Upwards of hundreds of thousands of nodes
  - Peta- to exabytes of data, gigabyte size files
  - Files striped across thousands of devices

- HPC workloads are demanding
  - Highly Parallel
  - Bursty, flash crowds, short inter-arrival times
  - Large, long lasting I/O

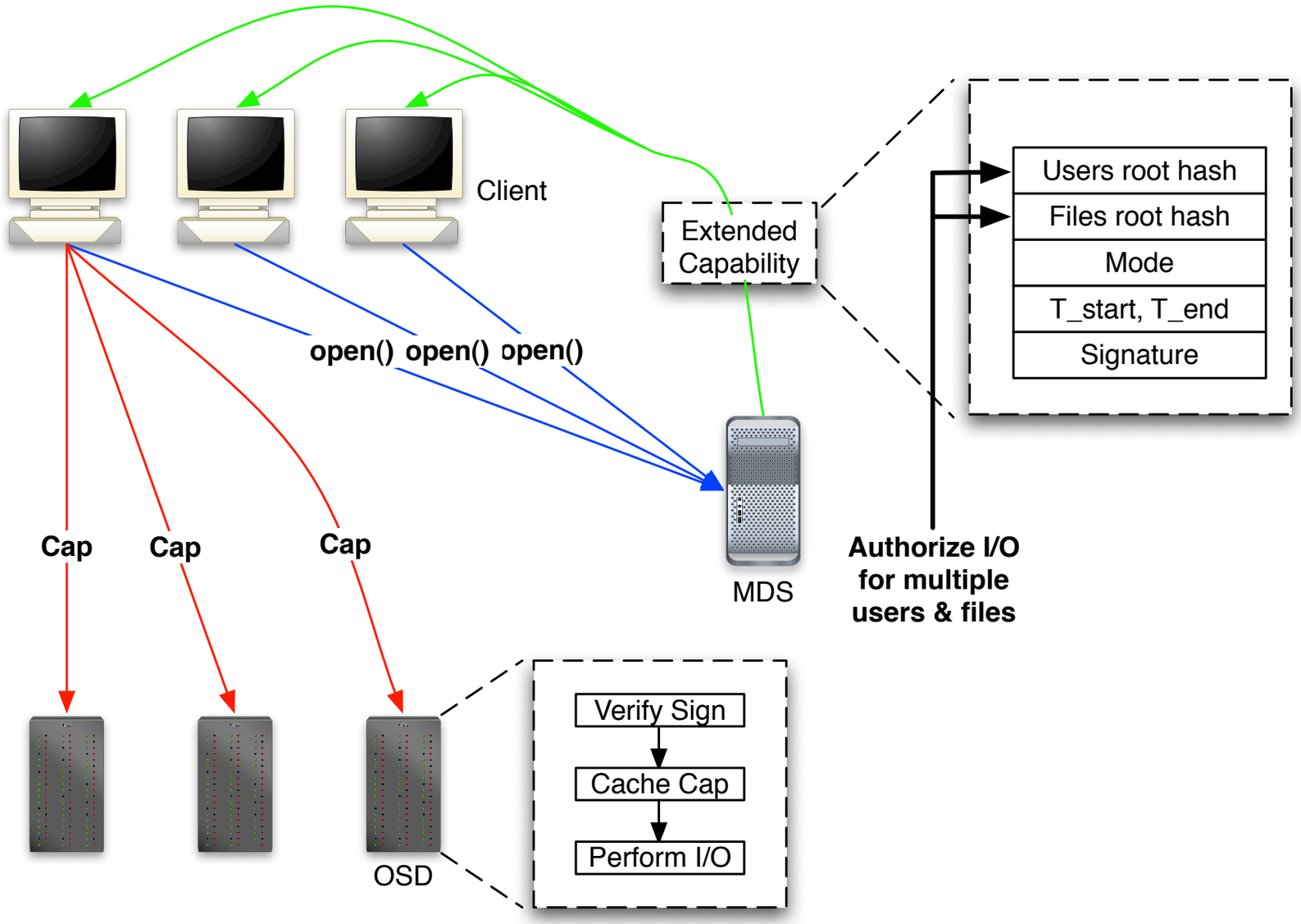
- How do we scale security for such a file system?
  - **Maat** - security for big, parallel file systems

pdsi

iSSDM

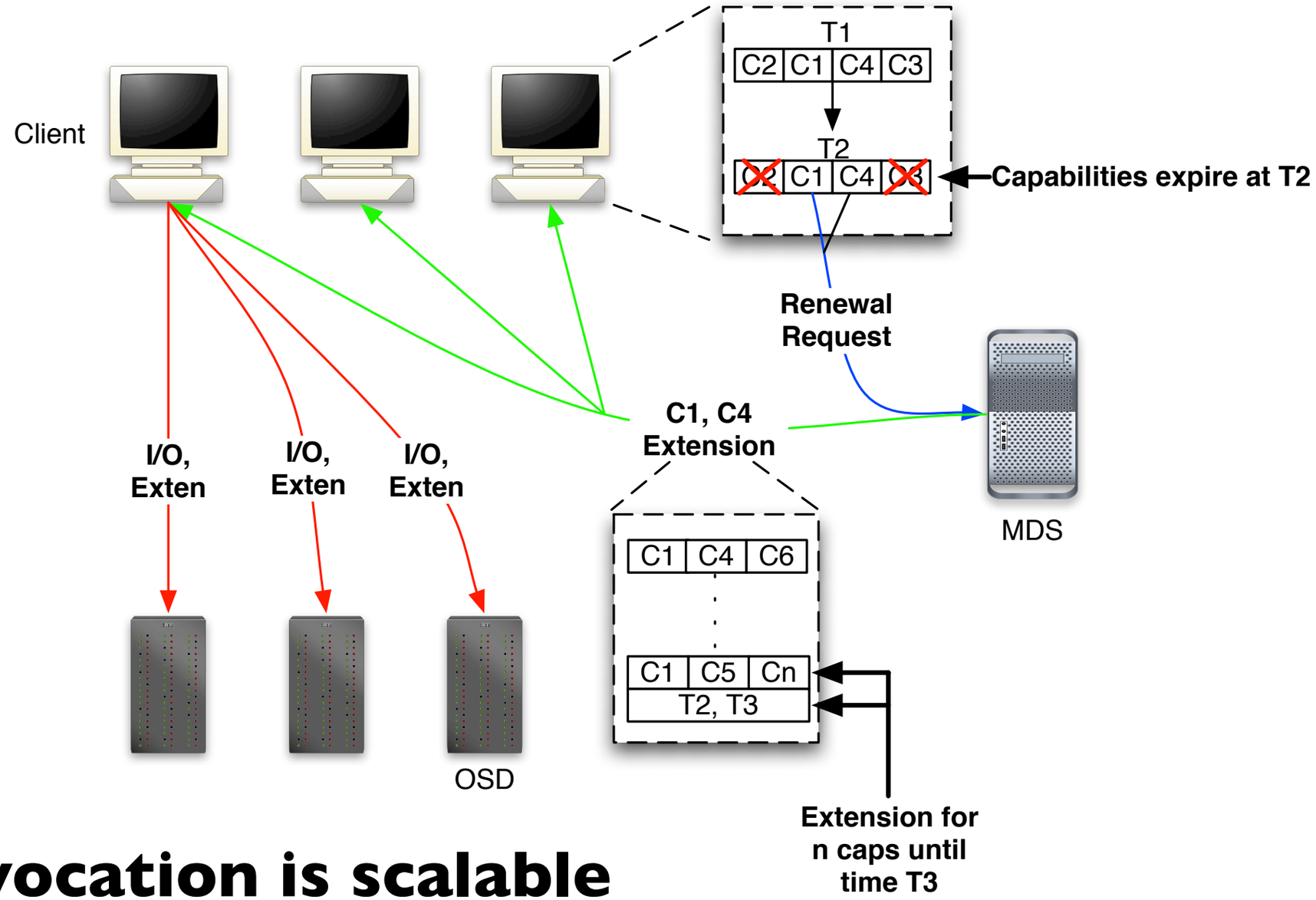
Storage Systems Research Center

Extended Capabilities



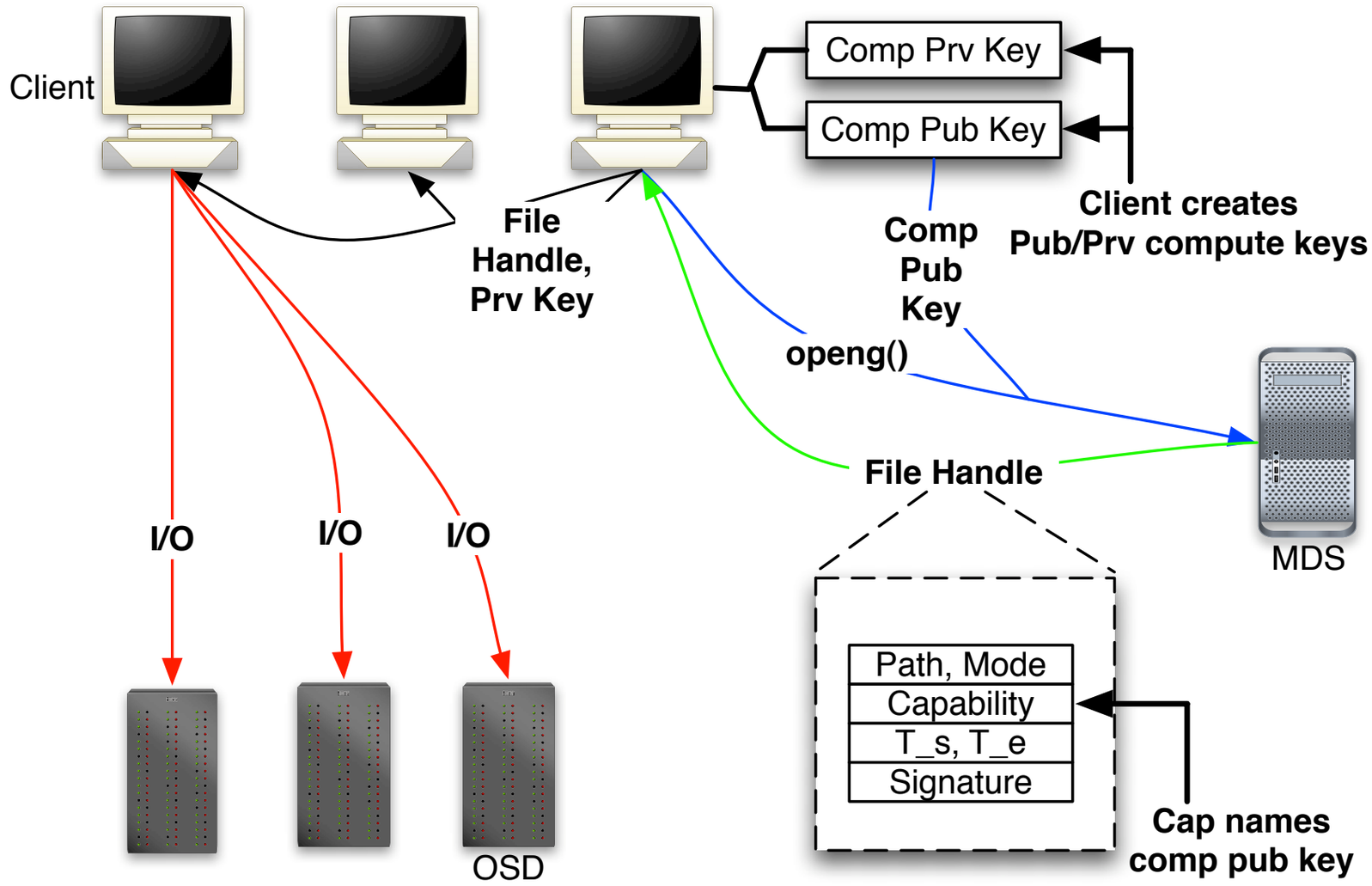
- **Reduces capability generation**
- Authorize I/O for any number of users and files
- Secured w/ asymmetric cryptography
- Enforces confinement w/ Merkle hash trees

Automatic Revocation



- **Revocation is scalable**
- Capabilities have short lifetimes
- expiration = revocation
- Shift problem from revocation to renewal

Scalable, Secure Delegation



- **Secure group computation**
- Open a file on behalf of many
- Delegate key pair rather than capability alone
- POSIX I/O extension: openg() and openfh()

Status

- Initial design discussion in an earlier paper
- Being implemented in Ceph petascale, parallel file system
- Future work:
  - Scalable on-disk security
  - Explore untrusted remote storage
- Questions?